Thesis for the degree of Doctor of Philosophy

Hunting Tools Beyond the Driplines

Performing large-scale nuclear physics experiments

Håkan T. Johansson

Department of Fundamental Physics
Chalmers University of Technology
Göteborg, Sweden 2010

Hunting Tools Beyond the Driplines
Performing large-scale nuclear physics experiments
Håkan T. Johansson
ISBN 978-91-7385-360-6

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 3041
ISSN 0346-718X

Department of Fundamental Physics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: +46 31 772 10 00

Cover:
Nuclear physics experiments are multi-disciplinary undertakings.

Chalmers Reproservice
Göteborg, Sweden 2010

Hunting Tools Beyond the Driplines
Performing large-scale nuclear physics experiments
Håkan T. Johansson
Department of Fundamental Physics
Chalmers University of Technology, 2010

Abstract

An important and growing sub-field of modern subatomic physics is the study of exotic isotopes using radioactive beams. Experiments started out simple, but are becoming more and more advanced. The increased complexity makes the use of software at all levels, from data acquisition to analysis, evermore present. However, it is rather the vast amounts of data collected in each experiment, than the sophistication of the calculations needed, that mandates the use of computers.

This thesis describes methods to unpack and handle raw data acquired in nuclear physics experiments, and discusses the importance of transparency in the data formats to allow access to the information also after long times. The major part of the work has been performed at the ALADiN-LAND setup at GSI, in conjunction with experiments studying halo nuclei. Aspects of data collection and issues related to network-based event-building are discussed and resolved. It outlines a possible mechanism extending the existing MBS data acquisition system to allow injection of persistent slow-control information, stored along with the events. At the detector level, it discusses the calibration and reconstruction of a device where the digitised signals need multiple levels of transformations before the physical interactions are recovered. Moving outwards to the level of analysing data for the entire setup, an ongoing work to develop an experiment-independent scheme for tracking of particles in a barely overdetermined setup is described.

Experimental findings for the very exotic ^{9,10}He and ^{12,13}Li systems beyond the dripline are shown, where ^{12,13}Li have been observed for the first time in this experiment. The intriguing results obtained earlier at the same facility for ^{7}He showing initially a weakening of the spin-orbit splitting have been reexamined using additional experimental information.

Keywords:
Nuclear Physics, Data Acquisition, Calibrations, Analysis, Unbound Nuclei, Radioactive Beams

List of publications

The following Papers are included in this thesis:

I H.B. Jeppesen, A.M. Moro, U.C. Bergmann, M.J.G. Borge, J. Cederkäll, L.M. Fraile, H.O.U. Fynbo, J. Gómez-Camacho, *H.T. Johansson*, B. Jonson, M. Meister, T. Nilsson, G. Nyman, M. Pantea, K. Riisager, A. Richter, G. Schrieder, T. Sieber, O. Tengblad, E. Tengborn, M. Turrión, and F. Wenander. Study of ^{10}Li via the ^{9}Li(^{2}H,p) reaction at REX-ISOLDE. *Physics Letters B*, 642(5-6):449 – 454, 2006.

II Yu. Aksyutina, *H.T. Johansson*, P. Adrich, F. Aksouh, T. Aumann, K. Boretzky, M.J.G. Borge, A. Chatillon, L.V. Chulkov, D. Cortina-Gil, U. Datta Pramanik, H. Emling, C. Forssén, H.O.U. Fynbo, H. Geissel, M. Hellström, G. Ickert, K.L. Jones, B. Jonson, A. Kliemkiewicz, J.V. Kratz, R. Kulessa, M. Lantz, T. LeBleis, A.O. Lindahl, K. Mahata, M. Matos, M. Meister, G. Münzenberg, T. Nilsson, G. Nyman, R. Palit, M. Pantea, S. Paschalis, W. Prokopowicz, R. Reifarth, A. Richter, K. Riisager, G. Schrieder, H. Simon, K. Sümmerer, O. Tengblad, W. Walus, H. Weick, and M.V. Zhukov. Lithium isotopes beyond the drip line. *Physics Letters B*, 666(5):430 – 434, 2008.

III N. Ryezayeva, C. Bäumer, A.M. van den Berg, L.V. Chulkov, D. Frekers, D. De Frenne, E.-W. Grewe, P. Haefner, E. Jacobs, *H. Johanson*, Y. Kalmykov, A. Negret, P. von Neumann-Cosel, L. Popescu, S. Rakers, A. Richter, G. Schrieder, A. Shevchenko, H. Simon, and H.J. Wörtche. Search for a low-energy resonance in ^{7}He with the ^{7}Li(d,^{2}He) reaction. *Physics Letters B*, 639(6):623 – 628, 2006.

IV Yu. Aksyutina, *H.T. Johansson*, T. Aumann, K. Boretzky, M.J.G. Borge, A. Chatillon, L.V. Chulkov, D. Cortina-Gil, U. Datta Pramanik, H. Emling, C. Forssén, H.O.U. Fynbo, H. Geissel, G. Ickert, B. Jonson, R. Kulessa, C. Langer, M. Lantz, T. LeBleis, A.O. Lindahl, K. Mahata, M. Meister, G. Münzenberg, T. Nilsson, G. Nyman, R. Palit, S. Paschalis, W. Prokopowicz, R. Reifarth, A. Richter, K. Riisager, G. Schrieder, H. Simon, K. Sümmerer, O. Tengblad, H. Weick, and M.V. Zhukov. Properties of the ^{7}He ground state from ^{8}He neutron knockout. *Physics Letters B*, 679(3):191 – 196, 2009.

V *H.T. Johansson*, Yu. Aksyutina, T. Aumann, K. Boretzky, M.J.G. Borge, A. Chatillon, L.V. Chulkov, D. Cortina-Gil, U. Datta Pramanik, H. Emling, C. Forssén, H.O.U. Fynbo, H. Geissel, G. Ickert, B. Jonson, R. Kulessa, C. Langer, M. Lantz, T. LeBleis, K. Mahata, M. Meister, G. Münzenberg, T. Nilsson, G. Nyman, R. Palit, S. Paschalis, W. Prokopowicz, R. Reifarth, A. Richter, K. Riisager, G. Schrieder, H. Simon, K. Sümmerer, O. Tengblad, H. Weick, and M.V. Zhukov. The unbound isotopes ^{9,10}He. Submitted to Nuclear Physics A.

VI K. Mahata, *H.T. Johansson*, S. Paschalis, H. Simon, and T. Aumann. Position reconstruction in large-area scintillating fibre detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 608(2):331 – 335, 2009.

U *H.T. Johansson.* The UCESB unpacker generator,
Long write-up — documentation and manual. Unpublished, 2010.

Contents

Preface – boot sector

This adventure started in the summer 2002 when I participated in an experiment at REX-ISOLDE, CERN, investigating the structure of the ^{11}Li unbound subsystem ^{10}Li with the ^{9}Li(d,p)^{10}Li reaction. It was quite overwhelming, but I do remember calculating the Rutherford scattering cross section in inverse kinematics. More fascinating was tracing all the cables of the data acquisition during one night-shift, making a paper drawing looking more spaghetti-like than the cables themselves. After the third or fourth re-drawing, the picture of how the data was collected became clear. (It was a FERA system.)

My arrival at GSI was early 2003, with a mission: analyse S245. The S245 experiment, performed in 2001 at the ALADiN-LAND setup, aims at investigating the structure of halo nuclei, by letting ^{8}He, ^{11}Li, and ^{14}Be break up on a liquid hydrogen (pure proton) target at relativistic energies. This is indeed a daunting task. Soon after coming to Darmstadt, a big announcement was made in the main auditorium: the future project of GSI (later dubbed FAIR) had been approved. Since then, I have in all ways conceivable tried to not only analyse S245, but to perform it such that the developments are future-proof.

Later that year, visiting KVI to participate in an experiment to investigate the unbound ^{7}He via the ^{7}Li(d,^{2}He)^{7}He charge-exchange reaction, introduced me to the beautiful data that gaseous detectors can provide; in this case the EuroSuperNova detector. Being able to help with the analysis software has led to some further involvement, later participating in another experimental campaign at KVI.

During 2004, the ALADiN-LAND experimental setup at GSI moved from Cave B to Cave C. At the accompanying preparations for the S287 experiment scheduled for the spring 2005, I simply could not get enough of working with the data acquisition system (DAQ). As the system was anyhow being ripped apart, it was a golden opportunity to learn it from scratch. Thus, some parts of the acquisition programs were re-organised. The most important development was the introduction of a total cable documentation system, including a time-consuming crawl of all 3000 cables.

2006 saw the inception of a tool to bridge the gap between the raw data of a DAQ and common analysis tools — named `ucesb` (unpack and check every single bit). It started as a reaction of disgust with the semi-manual unpacking methods at hand and is an orgy in the software concepts of lexers, parsers and templates. It has since become quite a neat pocket knife for raw data in all phases of an experiment. The success depends on hiding the complexities by a simple language for specifying the data layout of acquisition modules.

The current manual and description of `ucesb` is included as the last Paper of this thesis. It is a continued joy to develop and sharpen this software gadget, aiming to do more with less.

In mid 2006 work was started to supply the reborn analysis tool (`land02`) of the ALADiN-LAND setup with a generic tracker system, suitable for experiments with just barely over-determined tracks and a multitude of detector types. In the mean-time, the sheer ambition of and planning for a generic tracker has also enforced healthy design criteria for the parts of the analysis leading up to it. In particular, this has been manifested in the form of uniform handling of all charged-particle detectors and the use of error estimates of all determined hit coordinates (in space, time and amplitude).

With the move to Gothenburg early 2007, the following years have seen continued work on these and other things required to efficiently collect and analyse experimental data. Some of them are described in the latter half of this thesis. It is a sign of the times where nuclear structure experiments have reached a complexity similar to particle and relativistic heavy-ion experiments, necessitating a higher degree of specialisation. The first five included papers on physics outcome and allusions to them are mainly to show what the developments form a basis for — I have not calculated one single invariant mass spectrum. The more so have my colleagues. The perhaps best token of their appreciation of the capabilities of the tools behind the scene (`land02`) is the inclusion of me as first author of the upcoming Paper V.

Located towards the end is a Glossary, among other terms also containing alphabet-soup acronyms and abbreviations used in the text. Mnemonics and short descriptions of the detectors of the LAND setup are collected in Appendix A.

This thesis continues the tale of "The DAQ always runs" [1], and is organised according to:

"Sir, we are surrounded!"
"Excellent; we can attack in any direction!"
Gen. Oliver P. Smith (unconfirmed)

Chapter 1

Introduction

Nuclei, the cores of atoms, are femtoscopic systems — the typical length-scale is a femtometer, or fermi: 10^{-15} m — consisting of neutrons and protons. Each nucleus is far away from its neighbours, the distance given by the atomic length scale ångström: 10^{-10} m. As a comparison in two dimensions, humans have a typical scale of one meter. A similar separation would require your closest neighbour to be 100 km away — a population density of 0.0001 inhabitants/km^2 — likely not even achievable in small boats at the high seas.

This separation makes each nucleus serve as a laboratory for few-body quantum mechanics. But in contrast to atoms, which served as a nursery for the weird logics of quantum mechanics in the beginning of the last century, the description of nuclei from the outset pose much more difficult problems. An atom is characterised by the interactions of its electrons, mediated through the electromagnetic force. At the energies involved, the core of the atom is inert, and simply provides a positive electric field as a central potential. This field is successively screened by the electrons. Three more ingredients to take into account are the Pauli exclusion principle, preventing two particles (here electrons) from occupying the same quantum state, the half-integer spins of the electrons, making each orbital have two states, and special relativity.

The atomic cores, nuclei, are different beasts. There is, particularly for smaller nuclear systems, no central potential around which to efficiently and elegantly perform quantum mechanical calculations. The nucleons must themselves provide the potential in which they can be bound. The long-range electromagnetic force is only responsible for a repulsive term between protons in the energy economy of a nucleus. Binding is provided by the strong force acting among and between both the component particles: neutrons and protons. But the strong force has a socially more difficult nature than electromagnetism: far apart (on the order of a few nucleon radii) it quickly falls

to zero. To be provided with an attractive potential, the participants must come close. Upon coming too close however, the force reverses to be very repulsive.

1.1 Enter (extreme) isospin

The strong force is not sensitive to electric charge. Neutrons and protons behave the same under its influence. They are however different, described by having different projections of a quantum number called isospin: $\frac{1}{2}$ for the proton and $-\frac{1}{2}$ for the neutron[1]. The isospin projection of an entire nucleus is the sum of the nucleon isospins.

The distinction between protons and neutrons is important for the binding of nuclei, since it allows us to partially "circumvent" the Pauli principle, as one quantum number differs between the two kinds of nucleons. They can now be regarded as two sets, which can overlap much more. This is necessary, as the strong force is not able to keep just one set (n or p) of nucleons together[2].

The additional repulsive electromagnetic term for the protons make their orbitals energetically less favourable compared to the corresponding neutron orbitals. As nuclei become heavier, it is therefore advantageous to fill up more neutron than proton states, making the stable configurations for each mass number have a successively larger fraction of neutrons.

With increasing absolute value of the isospin, i.e. large differences in the number of protons and neutrons, the packing of protons and neutrons close enough for the strong force to provide binding starts to fail. Eventually, too many nucleons of the same type are present and will not be bound at all, marking the driplines of the nuclear chart.

Some nuclei at the dripline brink of nuclear stability exhibit a halo structure, where loosely bound valence nucleons can have a large fraction of their wavefunction extending into the classically forbidden region.

1.2 The shell model as reference

By mapping the chart of nuclides, measuring the two-proton and two-neutron[3] separation energies, it was found that for nuclei with certain numbers (dubbed

[1] Who is who does not matter, as long as the usage is consistent. With only two choices, nuclear and particle physicists originally still managed to use different definitions...

[2] With the exception of neutron stars, where gravity comes to the rescue.

[3] Steps of two are taken to avoid the effects of nucleon pairing.

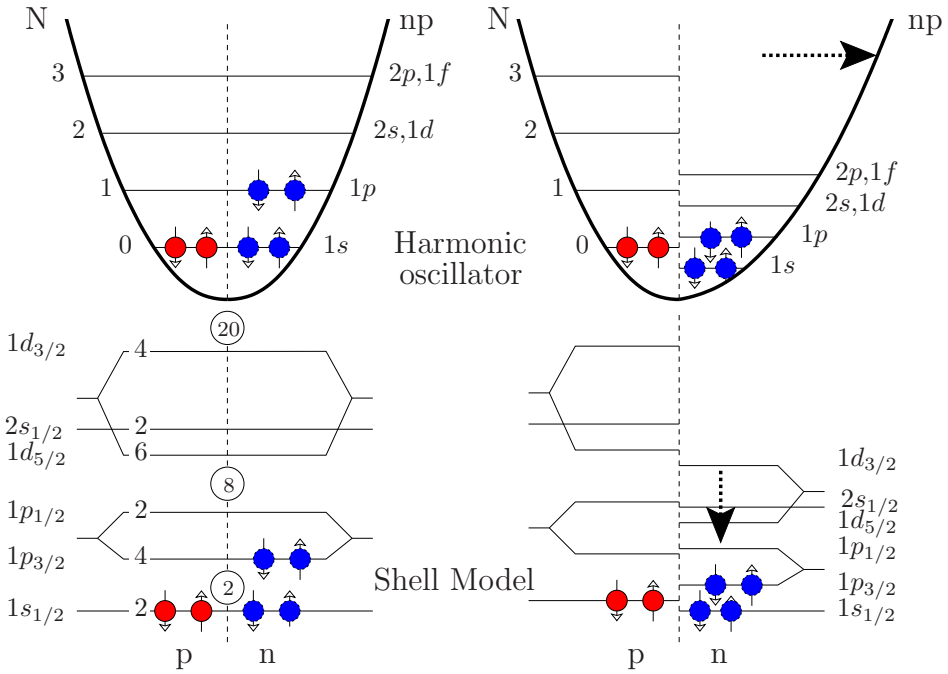

Figure 1.1: Degenerate levels in the harmonic oscillator are split by the more realistic Woods-Saxon potential and the spin-orbit coupling used in the shell model. The magic numbers are shown in circles. The diagrams to the right illustrate adjustment of the neutron orbitals in a halo nucleus, extending the potential well to larger radii. The filled circles represent the arrangements of protons and neutrons of ^{6}He within the models.

magic) of protons and/or neutrons the separation energies are signficantly reduced. These two-particle separation energies otherwise gradually increase as one moves along in the nuclear chart. This can be viewed as the nucleon states being grouped in shells, and that the nuclei with large separation energy correspond to configurations with completely filled shells. Adding extra nucleons to such a configuration leads to a drastic reduction of the corresponding two-nucleon separation energy. An analogous feature had already been observed in atoms for the ionisation energy. A successful theory needs to account for this effect and for the observed magic numbers.

The magic numbers (2, 8, 20, 28, 50, 82, 126) are not reproduced by the level-gaps within a harmonic oscillator model of the nucleon states. The shell model [2, 3, 4] differs from the simple harmonic oscillator by two modifications that break the degeneracy of the energy levels; a more realistic radial

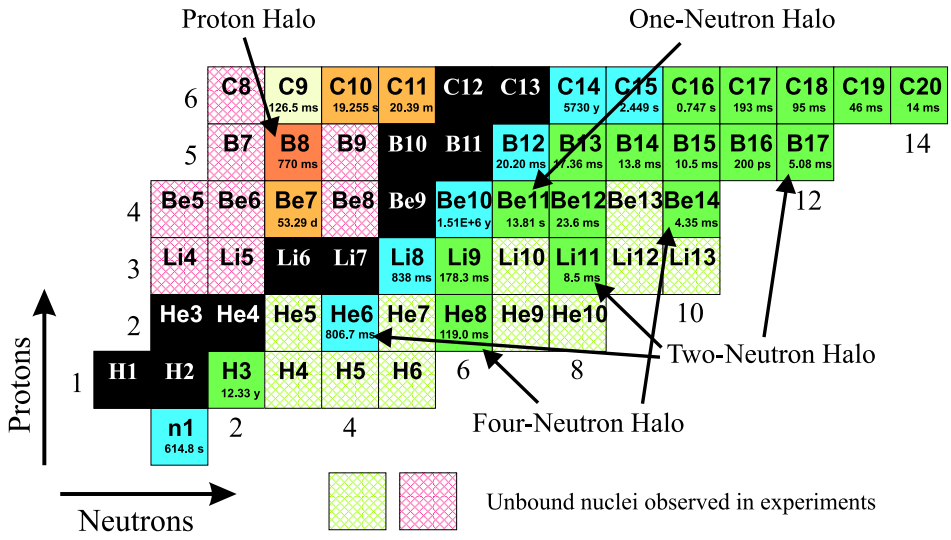

Figure 1.2: Chart of light nuclei. The jagged edge of the bound neutron-rich nuclei show a striking effect of pairing. From [5].

potential in the form of a Woods-Saxon function is used, and a coupling between the orbital angular momentum and the spin of each particle is introduced. This spin-orbit interaction can be constructed such that the magic numbers are correctly reproduced, see Figure 1.1.

The shell model provides a reference language of single-particle states that is used to describe nuclei. In fact, the shell model is so successful that the results of experiments, even in regions where its applicability is doubtful, are expressed in terms of how well the actual wavefunctions map onto the ones describing single-particle states, so called spectroscopic factors.

At the outskirts of the nuclear chart, close to the driplines, signs of "new" magic numbers have been seen, i.e. situations where the level diagrams are adjusted such that the large gaps occur for other numbers of nucleons.

1.3 Light nuclei – at and beyond the drip-line

Nuclei can reveal different properties depending on their position in the nuclear chart. Regions where aspects of the quantum mechanical description can be simplified are more accesible to theory, while regions where unusual phenomena are observed can serve as benchmarks.

The low-Z part of the nuclear chart is depicted in Figure 1.2. The experiments performed within this thesis work are concerned with light neutron-

rich nuclear systems consisting of two, three or four protons and up to ten neutrons. The abundance of neutrons makes them very loosely bound.

It is energetically favourable for like nucleons to pair up with opposite spins in the same orbital. Pairing affects the total binding energy, and its effect can e.g. be seen directly in a chart of nuclear masses. At the limits of nuclear stability pairing becomes indispensable to the very existence of nuclei, demonstrated by the jagged edge of the nuclear chart in Figure 1.2, and in extreme cases giving rise to Borromean systems [6].

The halo nucleus ^{11}Li consists of a mostly inert core of ^{9}Li with two halo neutrons, as does ^{6}He, here with an α core. These are Borromean systems in that they form a bound state although neither of the binary subsystems are bound: (1) the core and one halo neutron (^{10}Li or ^{5}He) nor (2) the di-neutron (n-n). They are good examples of how just a few nucleons can form an extremely intricate system.

Even 25 years after their discovery [7], halo nuclei present "nuts hard to crack" for both theoreticians and experimentalists. Various kinds of experiments allow to reveal different pieces of the puzzle, approaching a more complete picture. As different facilities and setups have different advantages, the various experiments are performed at different locations. Here will be described advances made at GSI (Darmstadt, Germany), at ISOLDE (CERN, Geneva, Switzerland/France), and at KVI (Groningen, the Netherlands).

1.4 The road from hardware to spectra (out of the ashes — into the fire)

Nuclear physics experiments are collaborative undertakings. In the research this thesis is concerned with, ten to twenty people are often involved at an experiment itself, to prepare the setup and collect the data. This does not account for the numerous technical staff needed to run an accelerator facility needed to produce the ion beams of interest. Analysis of the collected data is usually done by a handful of people — often taking several years for one experiment.

Many kinds of knowledge and experience in various aspects of nuclear physics (theory as well as practice), mechanics, geometry, electronics and computing are needed. As depicted in Figure 1.3, their application is often tightly intertwined and the work is interdisciplinary. Experiments stand on three pillars: physics, hardware, and software. They require each other in a borromean fashion. What follows is a list from an experimentalist's point of view.

1.4.1 Hardware

An experimental setup is made up of many pieces of advanced equipment[4]:

Radioactive beams are often mandatory to study exotic nuclei. As they are short-lived and only can be produced in miniscule amounts, they cannot be collected and machined into targets. They are instead created in one way or another upstream in the accelerator complex, and directed towards the experimental station.

Accelerator facilities of various kinds are used to create the ion beams needed. Depending on the experimental needs in terms of energy and isotopic purity, different schemes of production are used, see Figure 2.1.

Ions delivered to the experimental setups are either of the species of interest, or just one reaction away — to hopefully take place at the target — the heart and essence of the exercises. Some experiments measure and characterise reactions. Radioactive ions can also be collected on thin foils, to wait for and study their decay. Or they can be trapped to investigate other ground state properties, like the mass or magnetic moments.

Energy loss is the key to detection of *charged* particles. When ions, protons, electrons, muons, charged pions and kaons pass through matter, they loose energy, mainly through electromagnetic interactions with the electrons of the material. The energy loss is described by Bethe's stopping power formula:

$$\frac{dE}{dx} = -\frac{4\pi}{m_e c^2}\frac{nZ^2}{\beta^2}\left(\frac{e^2}{4\pi\varepsilon_0}\right)\left(\log\frac{2m_e c^2\beta^2}{I(1-\beta^2)} - \beta^2\right), \qquad (1.1)$$

where Z, β and E are the charge number, velocity and energy of the particle, the latter connected via $E = \gamma m_0 = m_0/\sqrt{1-\beta^2}$. The distance traveled is x, through a material with electron number-density n and mean ionisation potential I. The other are fundamental constants. By using materials (including gases) with suitable properties, the lost energy will not only contribute to heating and structural damage. It can also be available as e.g. scintillation light (photons), liberated electron-hole pairs in a semiconductor or ionised atoms or molecules in a gas.

Detectors are then such matter, together with mechanisms suitable to amplify and register the therein liberated secondary tell-tale particles. The

[4]Some of the first items listed may be small, and more of consumables...

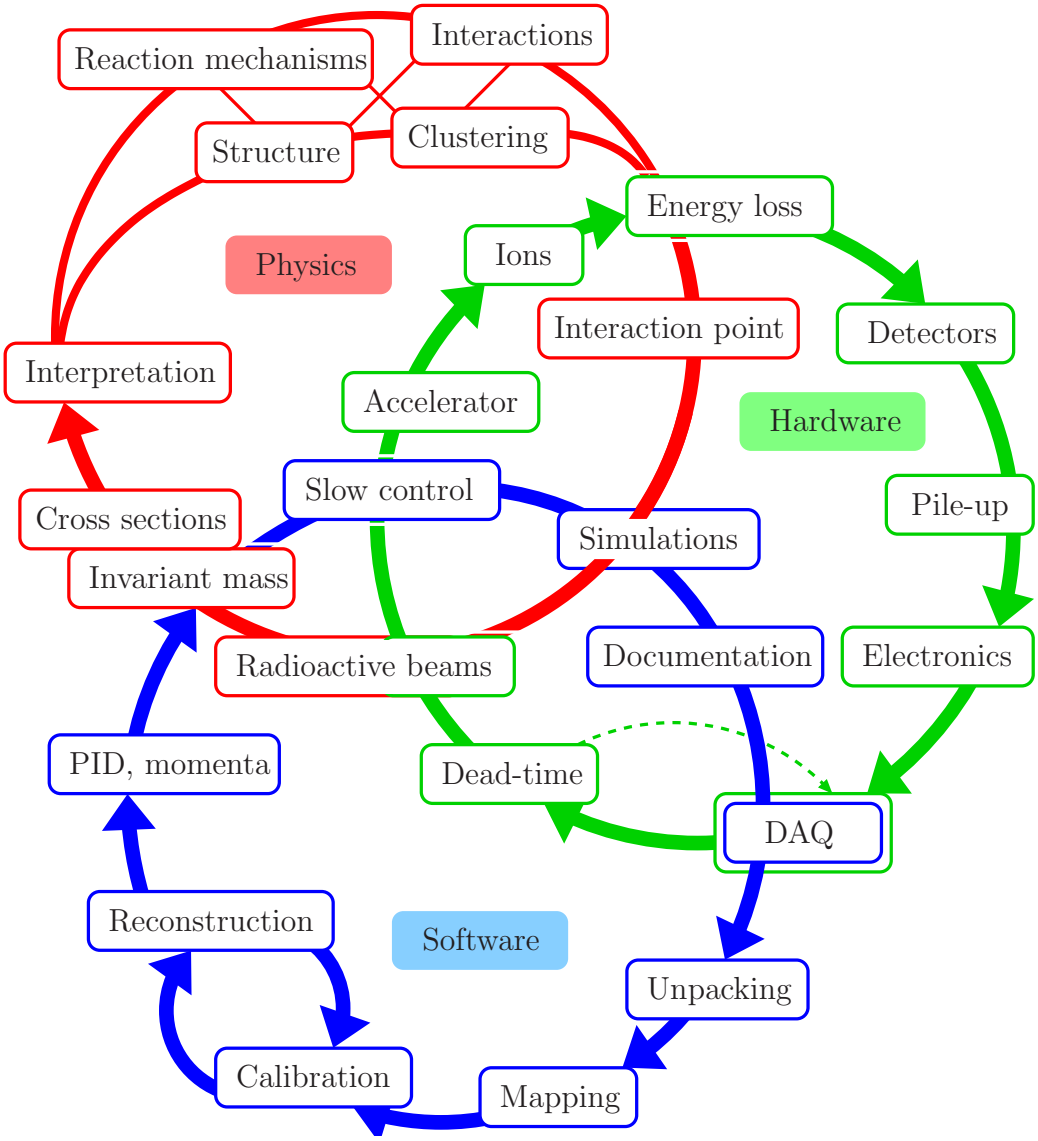

Figure 1.3: Borromean rings interconnecting ingredients needed when performing nuclear physics experiments. Persuading nature to reveal the inner workings of nuclei is an interwoven multidisciplinary undertaking: The underlying physics is probed as beams of ions impinge on targets. The resulting debris is characterised by detectors — by using all conceivable physical effects of radiation-, ion- and particle interactions with matter. Software is used to operate and control the hardware equipment. The raw information produced and collected by the DAQ is managed and processed by a multitude of programs. Observables to be compared to theory are laboriously extracted.

amplification and collection is done by e.g. PM tubes, attached electrodes or sense wires. To allow for later, computer-based analysis, the signals must at some point become mediated as electric signals, which can be digitised and stored.

Detectors are placed in the beam-line, or other areas where ions and reaction products should be detected. Charged particle detection via energy loss is relatively straight-forward. The functioning of the detectors themselves is governed by atomic physics or solid state physics. Ideally, an experimental physicist would like nuclear physics to only become involved at one single place — the reaction or decay under study itself[5]. The energy scales for atomary interactions (a few eV) are so small that the results (i.e. measured values) are statistical and can be treated as continuous variables. Whenever nuclear effects meddle in the detection process, the effects are abrupt. Thanks to the involved cross-sections and material thicknesses, it only happens in a small fraction of the events, but determining in which and where is not easy, when at all possible.

Nuclear and energetic electromagnetic interactions are however a necessary, but obnoxious and wicked associate in the detection of uncharged particles — neutrons and gammas[6]. Charged particles must be generated to in turn evoke electric pulses. The principles of gamma detection only marginally falls within the nuclear realm, being electromagnetic, but the energy scales involved are typically far above atomic scales. Electrons are liberated via the photo-electric effect, Compton scattering, or pair-produced with positrons. Neutrons only interact with other nuclei in matter, via the strong force. Their presence can only be unveiled at locations where they have scattered and thus transferred energy to charged nucleon(s), being detectable by the techniques applying energy loss described above. The detectors must therefore consist of material suitable both for the reactions producing charged particles, and detection of the same.

The treacherous nature of neutral particle detection also lies in its relative weakness. Detectors for charged particles are most often thin sheets a few hundred μm thick. The mean free path of a gamma or neutron at our energies — a few MeV and a few ten to a few hundred

[5]Exotic isotope production upstream in the accelerator facility is by nature an intense application of nuclear physics.

[6]While being birds of quite different feathers, with different interaction mechanisms of gammas and neutrons with the detector materials, many implications for the design of detectors are similar.

MeV, respectively — varies from a sizeable fraction of a dm to many dm, respectively. The detector volumes must be scaled in accordance to this, hence also making the localisation of the interaction points more difficult.

The basics of the detectors used in the ALADiN-LAND setup are described in Chapter 3.

Pile-up occurs when several particles hit a detector close enough in time, such that the two resulting electrical pulses are overlaid, thus invalidating the individual measurement of any of them. The time scales involved can vary from around a hundred ns for scintillators to several μs for semiconductors (due to the long shaping times required by the following amplification electronics). As normally only a maximum amplitude or integral is measured, it is generally impossible to detect the presence of pile-up from the affected detector alone. As pile-up spoils the measured values, analysis need to be performed such that affected events can be safely discarded, as the events otherwise are misreconstructed. Alternatively, the ion beam intensity must be lowered to reduce the relative probability of pile-up, or detectors must be built with many independent segments using more read-out channels.

Electronics, commonly in the form of standard function modules are connected together to form the physical and analog part of the body of an experimental setup. The detector signals continuously pass through the transformation stages, e.g. amplification, shaping and discrimination. A charged particle passing through a detector liberate tell-tale particles in the material (photons, electrons), the number of which are proportional to the deposited energy. A typical energy loss of 1 MeV would at complete efficiency and a presumed band-gap of 5 eV to be bridged for every release correspond to 200000 particles. At a pulse-length of 10 ns, this is a current of 3.2 μA. For commonly used signal cables with an impedance of 50 Ω, this corresponds to 0.16 mV. As electronics in general, and analog-to-digital converters in particular, work with potentials of a few V, amplification is clearly needed. The amplification must also be done close to the detectors, such that the tiny signals are not swamped in the inevitable and always present noise. For scintillators, the conversion from photons to electrons and amplification of the electrons is performed by photo-multiplier tubes at the light collection points at the edges of the scintillating slabs.

Based on coincidences, the trigger logics — acting as the hard-wired brain of a setup — determines when an event has happened (e.g. an ion

passing through the setup) and decides whether it should be recorded. The time available for decision making is usually below a few hundred ns.

DAQ — data acquisition hardware, is the central nervous system of the data recording. Digitisation modules (DAMs) are the final destination (and waste-basket) of the detector signals. At the occurrence of an accepted trigger, the characteristics of the signals (amplitudes, integrals and relative times) are held and digitised. The binary values representing the signals from each acquired event are transferred over various kinds of data buses to processor modules and finally stored on permanent media (tape and disk).

The signal sampling happens in parallel for all channels, and is a matter of at most few μs. Digitisation is usually done serially within each module, as the analog-to-digital converter chips usually are the critical parts, both with respect to price and power consumption. Conversion progresses in parallel for all modules, and can last for a few to several hundred μs, depending on the hardware. The read-out to the processor is serial for each processor in the setup, and the performance is usually balanced to take around the same time, or less, than the digitisation.

Dead-time is active when the DAQ hardware cannot accept a new trigger, either because it is currently sampling and digitising a previous event, or the output buffers are full. With a single-event read-out, the DAMs become full during each digitisation. Accurate handling of the system dead-time is required and critical to correct operation. In experiments that cannot rely on relative measurements, it must be precisely known for accurate determination of cross-sections and other absolute numbers. It is used to lock and block the trigger logics. Please refer to Chapters 5 and 6 of [1] for further digressions on dead-time, and how to combat it.

Another approach to alleviating the performance issues and penalties associated with a global dead-time generated from many components in large systems is the use of trigger-less schemes [8, 9]. Trigger decisions and event combination are done in software, in near real-time. They buy scaling simplicity by employing faster transport networks for digital data and front-ends capable of digitising all signals, not only those associated with triggers. Small pieces of dead-time can however remain in form of the recovery-time of individual channels as well as subsystem buffers becoming full and hence intermittently dead. These effects are pile-up like and may thus become a nuisance to analysis.

1.4.2 Software

Just as important as the physically easily identifiable hardware of a setup is all the virtually existing parts of an experiment. Software is needed to control and monitor the setup, and sophisticated tools are used to eventually extract the results.

DAQ software controls and sequences the read-out of the digitisation modules. It runs in real-time on the processor boards of the setup and is in charge of swiftly and efficiently write and transport the data to permanent storage, as well as distribution to on-line clients for monitoring. Depending both on the available beam intensities and the conversion speed of the DAM hardware, event rates of around a kHz, and up beyond a hundred kHz are common.

The data produced will be used for many years during analysis. As ambiguous and ill-defined information lead to painful analysis detours, prime factors of a DAQ design is simplicity and correctness.

Chapter 7 describes the experiment-independent and common stages of data transport implemented in the MBS DAQ system [10] and some modifications to make better use of the available hardware. Chapter 8 continues the discussion with a suggestion on how to implement injection and reliable storage of slow control information in the event stream.

Unpacking of the collected data is the first step of analysis. The values, as stored by the DAQ in files, preferably verbatim as delivered by the DAMs, must be retrieved and placed in data structures suitable for event-wise processing by analysis programs. This is needed for on-line monitoring as well as off-line analysis.

Chapter 5 explains why the unpacking step is important, and argues that it serves many purposes best as a separate link of the analysis chain. Being logically separate does not preclude tight integration with other tools in the chain.

Mapping the DAM channels to detector channels is a straight-forward task requiring knowledge of the cabling of the experimental setup. Data is event-wise copied from the DAM-oriented data structures filled during unpacking to data structures organised by detector. Some form of efficient sorting is generally required as part of the copy step, since module order often is pseudo-random, e.g. to avoid electronic cross-talk between neighbouring detector channels. It must also respect the

zero-suppressing primitives of both data structures. Zero-suppression is a vital part of data structures for modern, segmented detectors, since the number of channels that fire per event is much less than the total number of channels.

The attached `UCESB` manual (Paper U) describes an easy-to-use and powerful tool to perform unpacking, mapping and some generic monitoring tasks.

Calibration of all detectors of a setup is easily one of the most time-consuming and tricky parts of analysis. For example, when the signal amplitude of a detector channel is proportional to the energy lost by a passing particle, calibration parameters for transforming the digitised value to correspond to the energy loss need to be determined. The relative gains of all signal carrier conversions and amplifications, performed by the detector and electronics, are not needed. The digital value usually corresponds to a voltage measured by an ADC, in itself also uninteresting — as long as it is within range and overflow conversions are avoided. Only the overall factors are needed.

The difficulty lies in obtaining reference points. Some detectors can be calibrated with radioactive sources that emit α or γ particles with known energies. In many cases, such a procedure is impractical, or the energy ranges that can be covered with sources too small. One is then left with the daunting task to use the production data itself to also calibrate the systems. By making carefully chosen selections of events, based on their topology, e.g. criteria on hits in multiple detector-segments or detectors, some characteristics of one or a few of the hits can be assumed, and thus used as reference.

For time measurements, similar problems must be solved. Times are often measured as the voltage over a capacitor that was charged with a fixed current between the occurrence of two logical pulses. One of the logical pulses is often common to the entire setup and derived from the trigger. The value therefore corresponds to the (relative) time of the other logical pulse. First, the gain of the individual time values must be converted to a common time scale. Then, the times must be synchronised — adjusted relative to each other such that time differences between arbitrary channels can be meaningfully calculated. The first step is straight-forward by using special hardware which assists by creating events suitable for time calibration. Chapter 9 details a method to achieve absolute time gain calibration using a few standard electronic modules. The second step almost exclusively requires the

use of particle hit correlations in the various detector components.

Chapter 4 of [1] presents detector-internal calibration using ion beams and cosmic muons, and Appendices A and B therein discuss the importance of the choice of parameter representations.

Reconstruction happens event-wise and is relatively straight-forward. The input consists of the digitised values for an event and the previously determined calibration parameters, and as output more refined information is expected. The reconstruction progresses in steps, each taking the data to representations closer linked to the physical processes under study: hits at the detector level, assembled as reconstructed particles, then combined to evaluations of the properties of the nuclear events.

Reconstruction begins by transforming the time and amplitude values associated with a hit at some point of a detector into suitable coordinates and units. For position-sensitive devices the location can then be calculated, by taking the functional principles of it into account. Finally, the hits in several detectors are combined to identify ions and other particles traversing the setup. The step-wise nature of reconstruction is accentuated during boot-strapping of the analysis process where reconstruction progresses in tandem with calibration. As higher-level data becomes available with each step, data selections can be made to determine the parameters needed for the next stages of reconstruction.

Reconstruction is needed at all phases of analysis: on-line, near-line and off-line. Off-line analysis is the careful treatment of data that is performed after an experiment has been conducted. This has the highest demand on accuracy and handling of all special effects that occurred and affected the performance of the detectors during the experiment. On-line analysis is the data mincing performed for monitoring purposes on the live data, i.e. as it streams from the running DAQ. The results are continuously displayed in various spectra — histograms, scatterplots and ascii-art[7] — and can also be used for trending, showing the performance of the setup components as functions of time. As the mode of continuous streaming of data does not lend itself to interactive use and operation, the requirements on the software are towards smooth unattended batch operation.

On-line monitoring naturally requires calibration parameters. These are obtained by performing near-line analysis, i.e. by doing quick off-line analysis on the freshly recorded data. Near-line differs from on-line

[7] Using a computer terminal window in text mode to display the information. This is suitable for remote operation, and can be remarkably expressive.

in that it entails user interactivity. Relative to off-line analysis, especially for setups with many detector channels, its successful application depends critically on the availability of routines to perform the bulk of the work automatically. The reason is simple — the time-scales of days and weeks that can be spent for off-line treatment simply are not available during an experiment. The same calibration and reconstruction software can be used for all the listed phases of analysis. By aiming the design at fulfilling all the various requirements, especially those set by on- and near-line analysis, a more powerful monitoring is possible. This gives the chance to detect and correct or replace malfunctioning equipment early during data-taking, and thus often completely avoid the need for tedious off-line workarounds — or worse, if data becomes unusable.

Paper VI and Chapter 6 of the work at hand describe the calibration and reconstruction of a simple, yet intricate detector species.

Particle ID and **momenta** are the final products of the setup- and detector-oriented part of analysis. Information from all detectors is combined, and the particles are identified by e.g. their energy loss, time-of-flight and curved trajectories in magnetic fields. The momenta of the identified incoming and outgoing particles as they enter and leave the presumed interaction point in the target are also calculated. This combined information serves as a relatively clean interface to the subsequent physics analysis.

Slow control systems are used to set and monitor the many parameters that determine the operating state of a setup. High voltages, discriminator thresholds, timing constants, electric and magnetic fields, gas flows and mechanical actuators all need to be monitored and controlled. While computer control often is faster than screwdriver-operated potentiometers and allows for automated feedback-loops, the name slow control may stem from two characteristics: the controlled parameters should vary only slowly, if at all, during an experiment, and the need for speed are also very moderate in comparison to the data collection. Robustness and correctness are however paramount, as erroneous data (e.g. high voltages) can destroy equipment, or render collected data useless.

With larger sizes of setups, the need for unified control systems increases. It is impractical when every kind of device comes with its own control computer. Consolidation is rather easily achieved with commodity network and common user interfaces. As usual, dedicated

electrical interlocks must be employed in cases where network outages or other mechanical mishaps (unconnected connectors) could lead to hazardous situations.

Simulations are used to compare results to expectations. The main motivator is the ability to deal with non-uniform detector responses, complicated acceptance cut functions introduced by the setup geometry, and multiple scattering in the target or elsewhere.

While event reconstruction is something similar to equation solving, simulations are more of forward calculations. Comprehensive simulations can include detailed descriptions of the setup geometry and materials. They are generally of Monte-Carlo character. It should be noted that reconstruction needs a more abstracted (simpler) description of the setup geometry than simulations. As it has to deal with the data that was actually measured, and back-solve for the ion tracks, it is not of the same probabilistic nature. To be able to do the equation solving at all, the objects comprising the setup must be treated as larger effective aggregates, with less detail — often just one uniform block or sheet of material per detector.

Two regimes of simulation scopes can be identified — separated by length-scale and connected at the interaction point. One deals with the nuclear reactions of interest and must consider the underlying theory of nuclear structure and reaction mechanisms. Experimentalists tend to prefer this in the form of *event generators*, preferably provided by theoreticians. (This black-box practice is naturally despised and frowned upon by theoreticians.) The other regime deals with the propagation and detection of incoming and emitted particles through the macroscopic setup. Cross-sections of especially nuclear interactions are treated more sweepingly, the important features are the overall average distortions to the presumed "optimal" / unperturbed particle tracks.

The purpose of an experiment is reconstruction back to the interaction point, and from there also the recovery of information about the underlying principles of nature. However, the equation solving procedures usually break down at the point where a setup's geometry-induced acceptance cuts must be managed. Simulations are then used to reach this point starting from the other end, but then requiring the assumption of the underlying theory. The assumptions must then be varied until a matching is reached. The drawback of this approach is the strong dependence on all information that went into the forward calculations — the published results become hard to compare with other

theoretical models of the actual reactions studied. Back-calculation with reconstruction is disfavoured by the very large error bars it would introduce.

Documentation is a necessity. With experiments that can take 10 years through analysis, a setup that has operated for twice that time and a mean interaction time of the main work force — PhD students and post-docs — of less than half that time, it becomes paramount.

Appendices C and D of [1] describe a cable documentation system based on text-files. It performs consistency checks and generates the lists needed for data unpacking and mapping, thereby enforcing the documentation of most parts of a setup's electronics.

1.4.3 Physics

The goal of the analysis of experimental data is an interpretation of the results in terms of the physical laws and principles governing nuclei.

Invariant mass is a quantity preserved during coordinate system Lorentz transformations [11]. It is the main working principle of the ALADiN-LAND setup, when determining excitation energies of unbound systems. The states of interest can be created by Coulomb excitation, i.e. the absorption of a high energy (virtual) photon when passing by a high-Z nucleus [12, 13, 14], e.g. Pb. Or they can be the states of the remaining excited nucleus after removal of one or a few nucleons by nuclear diffraction or knock-out.

When the excitation energy is below nucleon separation energies, the state will usually decay by gamma emission and we would perform a gamma-ray spectroscopy experiment. For particle unbound states, the excitation energy will predominantly be reduced by the emission of nucleons since the strong interaction outweigh the electromagnetic. If the remaining nucleus has bound excited states, further gamma emission will occur. The decay energy thus either goes into relative energy between a nucleon and the remaining fragment or is emitted as γ-quanta (and a typically very small kinetic recoil energy of the emitter). These back-to-back emissions obey the laws of energy and momentum conservation. The calculation of the invariant mass, in any coordinate system, recovers the excitation energy of the original excited state (with its ground state energy or rest mass as reference). The calculation can also be performed with one of the ejectiles missing, in which case the full reaction kinematics of the incoming, target and its recoil particles

is used to recover the calculation. An example of this method are the reaction experiments at REX ISOLDE (see Paper I) where by design only a light ejectile (recoil) is measured, instead of the heavier excited fragment.

Cross sections tell how likely it is that one projectile shot at another will interact, in terms of the equivalent vulnerable area exposed by the target to an assumed point-like projectile. The cross section, being a macroscopically small quantity, is measured in barns (= 10^{-24} cm^2). Typical cross sections measured at the ALADiN-LAND setup are on the order of mb, what together with extremely small particle rates — not more than several hundred per second, would end up with very low statistics. But, high beam energies permit the use of thick targets, up to 1 g/cm^2, and that increases the reaction rate. At the same time, multiple reactions for each projectile nucleus have to be avoided, limiting the target thickness. Allowing reaction probabilities of 5 % is customary in the LAND setup.

Cross sections are important observables in experiments. They are often determined and given differentially, e.g. as function of incident energy and outgoing angle. In some cases (and models), the cross sections can be well described as the actual geometric areas subtended by the reacting nuclei, but quantum and structural effects may cause drastic modifications[8], whose determination are the objectives of our measurements.

Interpretation of the measured quantities into information about nuclear structure, constituents and forces forms the interface between experiment and theory. The measured quantities are typically not the observables obtained from theoretical descriptions of nuclei. One simple example is the measurement of the (apparent) size of a nucleus. It can be inferred from e.g. reaction cross sections, not from the use of a

[8]The former case would be analogous to trying to knock down tin cans with some slug ball at an amusement arcade. To determine the cross section experimentally, you would have to actually shoot without aiming and locate the tin cans uniformly over the area covered by the fire. This requirement is (unfortunately) immediately fulfilled by nuclear physics experiments, as the atoms in a target are uniformly distributed over the area which even a very well-defined beam covers. The cross section obtained from such an exercise should match the result of a geometrical calculations just taking the shape of the cans and slugs into account. Now assume that both slugs and cans are magnetic. The chances of interaction will be affected. This serves as the analogy of quantum mechanical effects. It is also easy to imagine that the correction effects are energy dependent, i.e. vary with the velocity that the projectiles have.

calibrated (femto)meter stick.

Reaction mechanisms play a crucial role in the interpretation of observables from experiments with interacting nuclei. Typically a reaction can proceed through a mixture of mechanisms. Direct and compound, knock-out, electromagnetic excitation and dissociation reactions, as well as quasi-free scattering are but a few [15]. To some degree the dominating mechanisms can be selected through the energy available in the centre-of-mass system and the choice of target material. At low energies the reactions are more collective, taking place at the surface of the colliding nuclei and involving valence nucleons. At higher energies the interiors can be probed.

The reaction mechanism and its affect on the results are crucial ingredients for all the experimental Papers included in this thesis (I – V). These effects can be considered either by handling them explicitly, or e.g. arguing that they are negligible. For unbound systems the final-state interactions, which could be considered as post-reaction mechanisms, actually cover the entire life-time of the studied nuclei.

Interactions between the constituents of matter give rise to its properties. Nuclei are formed and bound by the strong force, with corrections introduced by the electromagnetic force, more pronounced as the number of protons increase. The weak interaction mediates beta decay, which helps nuclei that have large excesses of neutrons or protons to reach a more favourable configuration, but is otherwise of no major concern to nuclear structure itself. The fourth fundamental force, gravity, can be ignored.

While electromagnetism, together with the weak interaction, is rather well understood, the effects and mechanisms of the strong interaction when binding nuclei is a tougher puzzle. The interaction of quarks and gluons at high energies, where they act more independently, is described together with electroweak forces by the Standard model. However, at the lower energies of nuclear binding, the interaction becomes non-perturbative and the basic degrees of freedom are nucleons — that are themselves complex objects consisting of quarks and gluons.

Many forms of effective models for the strong interaction are used when describing the strong force's hold of nuclei.

Nuclear structure describes how aggregates of protons and neutrons arrange themselves, under the influence of the strong and electromagnetic forces. However, different models of nuclear structure have to be used

in different mass-regions of the nuclear chart. Calculations of various properties using theoretical models constrain the allowable parameter space when the properties are converted into observables and compared to experiment. Some models start from rather basic principles — ab-initio, and attempt to build nuclei using realistic interactions. This only works for light nuclei [16, 17, 18]. Others are of a phenomenological character. The grand-old lady in the game is the shell model [2, 3, 4], which also provide the naming-scheme of nucleon orbitals.

For experimental studies, different nuclides simply provide different conditions, as this is the way to vary the number of protons and neutrons. Exotic isotopes, with large excesses of protons or neutrons, are useful as they provide unusual media for the interactions to work within — highlighting border-cases of nuclear structure.

Clustering is a way of arrangement which is preferred by nucleons in some circumstances. One example is halo nuclei, where a core of nucleons is surrounded by one or a few additional nucleons that are only loosely bound. The loose binding allows their wave-functions to be spread over an unusually large volume, thereby probing the effects of the strong interaction at larger distances than is the case in normal nuclei.

When clustering models pick out the most important degrees of freedom, aspects of nuclear systems can be investigated by studying them as a set of cluster components and their interactions. This simplification by the reduction of the number of bodies involved can be of great importance, e.g. when evaluating the effects of reaction mechanisms. Halo nuclei are systems that in many respects exhibit clustering. Another prominent example is the appearance of α-clustered systems, e.g. along the $N = Z$ line.

The **interaction point** is where physics meet the setup. The incoming ion, prepared and characterised by the accelerator and detectors plays the chance game of quantum physics. Outgoing ions are measured. Experimentalists aim to provide fair and transparent play for all ions, while theorists struggle to determine by what rules the game is actually played. If the ions could, they would likely just laugh...

Chapter 2

Nuclear physics experiments

Life as an exotic isotope on earth is hard. You are born an orphan, in a violent demolition reaction, as your only parent nucleus is fragmented or fissioned into leaving you, along with some other pieces, often protons and neutrons. Or possibly your two parent nuclei are fused, leaving you with a neutron sneaking away with the excess energy and angular momentum.

Life expectancy is also short, intimately linked with being exotic — stable isotopes can live forever. The nuclei are exciting as they represent unusual configurations of protons and neutrons. So odd and awkward that they themselves sooner rather than later rearrange, under the influence of the weak force, letting an excess neutron or proton decay into the other kind of nucleon along with the emission of an electron or positron and an (anti)neutrino.

Going one step further, beyond the frontier known as the drip-line, the nuclear configurations are not able to stay together for long enough to form bound states. They rather immediately fall apart, expelling the surplus nucleon(s). When viewing the process as trying to build heavier nuclei by adding consecutive neutrons or protons, the additions just drop off. The constituents perform the act of nuclear "ships passing at night". But unlike safe maritime operations, it is every time a close call — hard port or starboard. By measuring the trajectories of all outgoing particles, calculating the invariant mass and looking into angular and energy correlations we can infer how the nucleons briefly interacted.

2.1 Radioactive beams

Nuclei at the drip-lines of the nuclear chart have short life-times, usually a few milli-seconds, so they must be produced in the same twinkle-of-an-eye that they are subjected to investigation.

Creation of exotic isotopes requires nuclear reactions, with stable (or close to stable) isotopes and energy as ingredients. The stable precursor nuclei are brought to destruction by letting them collide violently with other stable nuclei. The large amounts of energy — concentrated into the immensely tiny nuclei a few fermi large — are needed to allow a stable nucleus to rearrange into the combinations of energetically less favourable compositions that can be found in the debris. Some of it as exotic nuclear species. Even fewer as the nuclides wanted in an particular experiment.

The production process through nuclear collisions can be organised in two different ways, depending on how the exotic isotopes shall be investigated, see Figure 2.1. When the rare isotopes are needed at rest (e.g. for decay studies) or at very small beam energies, it is most effective to let the heavier precursor nuclei start at rest, being hit by lighter projectile nuclei carrying the energy to the production target. The produced exotic atoms are gathered from the hot target by diffusion into an ion source. Following ionisation, they are electrostatically extracted and magnetically mass-separated with respect to A/q, with $q = 1$ as the ions are single charged. This is the isotope separation on-line (ISOL) method. The main drawback is the dependence on chemistry within the target and ion source to selectively extract and ionise only atoms of the wanted species (proton number, Z), provided an ion-source-type suitable for the element exist at all.

With experiments requiring rare isotopes at higher energies, it is easier to accelerate the nuclei that will be fragmented or fissioned and let them collide with a thick production target, distributing the products in the forward direction. The exotic species are thus created in-flight and can be separated directly by a magnetic spectrometer as they travel towards the experimental target. This method has the advantage of being independent on the chemistry of the elements produced. It is unsuitable for producing isotopes for experiments performed at rest (decay studies, trap experiments) or low energies, as the radioactive beams must be decelerated, leading to broad beams with large emittances.

The undertaking of reaction studies at low energies, e.g. transfer experiments around the Coulomb barrier, requires an additional acceleration after ISOL production (as in Paper I at REX-ISOLDE [19]), or deceleration (as envisaged for HI/DESPEC [20]). The beam energies involved are larger than achievable electrostatically at an ion source, but smaller than required for in-flight isotope production.

In some cases more traditional experiments with both stable beams and targets can be used, with the advantage of higher intensities. Using Paper III as an examples, the creation of the unbound system ^{7}He in the ^{7}Li(d,^{2}He)^{7}He

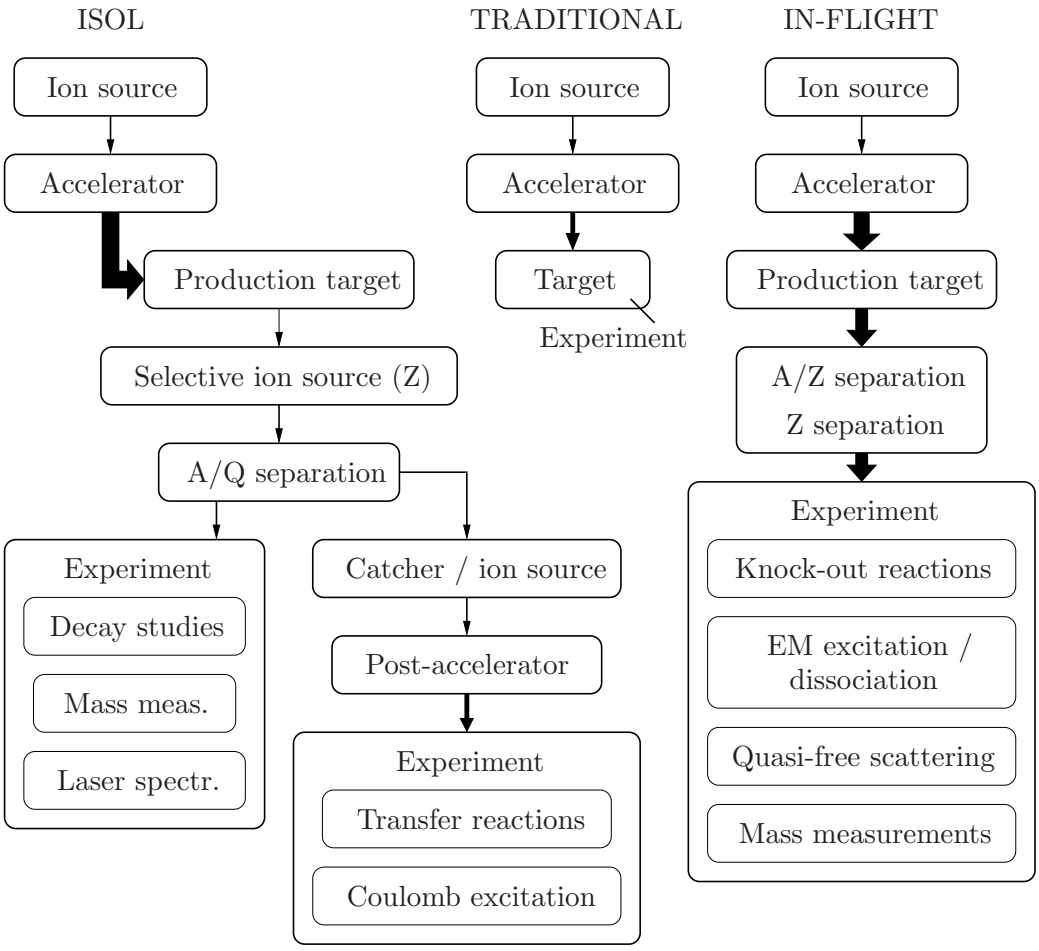

Figure 2.1: Outline of the ISOL and in-flight methods for rare isotope production and study. ISOL facilities produce and provide exotic isotopes at low energies with small (good) emittances. In-flight separation of nuclides from fragmentation and fission gives high-energy beams with large (poor) emittances. Less elaborate traditional methods with stable beams reacting with stable targets can be used for special cases, with the advantage of much higher rates at the experimental target. The thickness of the arrows roughly represent the energy of the ions in the respective beams.

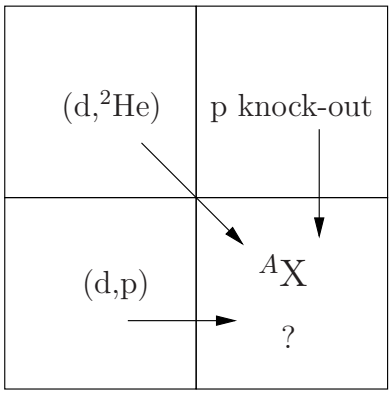

Figure 2.2: Exploring the nuclear system of interest from several different directions, using various probes and reactions. A nice picture of how ^{7}He has been attacked via many paths is given in [21].

charge-exchange reaction[1] uses only stable nuclei as target and beam.

2.2 Probes from different directions

In this thesis, reactions are described with the A(B,C)D notation, using the convention that A is the target nucleus and B is the projectile. The observed reaction products stand on position C and D is reserved for unobserved ones.

The properties (mass, spin, parity, magnetic and electric moments) of a state reflect the particular arrangement of the protons and neutrons. They can be calculated from the wave-functions — provided these are known. But theorists can only compare to a small, limited set of numbers — the experimental observables. And based on that, the models are adjusted to reproduce the experiment.

An experiment to determine properties of nuclei is designed to emphasise and extract one or a few parameters of it. This since the conclusions are drawn from data registered as count rates for some classes of outcomes relative to other. Often, the nuclei have to be probed violently enough to provoke fundamental changes in order to expose the property of interest. Figure 2.2 shows a few of the simpler reactions that can be used to study an isotope.

Unbound nuclei are no different. The method of probing also determines which system is studied, either the incoming or outgoing. In a transfer re-

[1]The term ^{2}He denotes an unbound system and is particular for that setup, referring to the detection of both protons; the reaction would normally be written (d, 2p).

action at low energies, e.g. ^{9}Li(d, p)^{10}Li as in Paper I, the resonances of the unbound system ^{10}Li can be populated, making it possible to determine the positions of its ground and excited states. By measuring the directions of the outgoing protons, the spin of the created states can also be inferred. The detected particle, the outgoing proton, together with the known input momenta, act as telltale of the remaining system (^{10}Li). There is however no guarantee that the colliding nuclei do what one would like to measure. Instead of the proton just acting as a spectator in the transfer reaction, the entire deuteron may interact with ^{9}Li as a whole, forming a compound nucleus, ^{11}Be in an excited state. If this decays by proton emission, it will blur the experimental results.

The energies used for different experiments are dictated by the kinematics of the reactions as is the placement and working principles of the detectors. Enough energy (but generally not too much, as that opens irrelevant and background-generating reaction channels) has to be available in the centre-of-momentum system of the reacting nuclei. The detectors are placed, in order to detect some or all products with maximum granularity (in space, time, energy loss). With high (instantaneous) rates of incoming ions, the geometry of a setup has to be designed to suppress detection of unreacted beam particles. This is the case for reaction experiments at REX-ISOLDE, where only a reduced number of the unreacted ions are detected, for determination of cross sections.

Knock-out reactions at high energies, e.g. removal of one neutron from ^{11}Li, also create ^{10}Li. Here, it is the remnants of the unbound ^{10}Li itself (usually ^{9}Li+n) that are detected, giving access to further internal variables like angular correlations, at the expense of worse determination of e.g. the excitation energy of higher resonances as compared to the transfer experiments.

In the same type of experiments, also the ^{11}Li system can be studied, by instead selecting events where the halo nucleus has disintegrated more gracefully into its subsystems, leaving both neutrons detectable at small relative energies. In the ALADiN-LAND setup lab-system frame this translates into, and is taken as, the forward direction cone defined by the LAND detector size and distance.

Chapter 3

The ALADiN-LAND setup

As an example of a nuclear physics experimental setup, the following will describe the ALADiN-LAND setup, slightly biased towards its appearance during the S245 experiment. While that actual beam-time predates my involvement, it is the setup I have worked with the most, performing analysis in the form of detector calibration and reconstruction. During half a dozen other experiments the last five years, the data acquisition system of the ALADiN-LAND setup has been the proving ground for many of my developments.

3.1 Detectors in general

The ions in the experiments (outgoing and usually also incoming[1]) need to be detected. In the experiments described in this thesis the measurements are done exclusively with detectors that transform the hits to electrical signals. See Appendix A for a list of the detectors used in the LAND experiments. We are not involved with (nowadays) exotic detectors using photographic emulsions, bubble chambers, etc...

There are different means to make a passing charged particle produce an electric signal that can be digitised and recorded by a computer. They are based on exciting some part of the detector material. Alas, the detection always requires the ion to lose some energy. This then leads to liberation and a small "flow" of some kind of particles (photons from de-excitation, electrons or ions) towards one or several collection regions. If the flow does not already consist of electrons (suitable for electronics), the collection mechanism also

[1]The incoming ions may only be attempted to be detected if they have so much energy that they can pass through a detector and still have enough energy left to be useful for the experiment afterwards, i.e. without severely impairing the beam characteristics.

needs to do that conversion:

- Photons are released from molecules excited by passing ions in a **scintillator**. They are detected and converted into electrons making use of the photo-electric effect in the entrance window of a PM-tube. The knocked-out electrons are accelerated by an electric field towards several stages of dynodes. At each dynode, each incoming electron (on average) knocks out several electrons, creating an avalanche, such that a multiplication of the current takes place, up to a factor 10^6.

  The signal can be directly fed into a QDC, integrating the charge. Common full scale ranges are on the order of a few hundred pC. A pulse 5 ns wide, with a total charge of 100 pC represents a current of 20 mA. The pulse height as measured over a cable impedance of 50 Ω is then 1 V.

- The electron-hole pairs created by a passing ion (or a photon depositing its energy) in a **semiconductor** detector (silicon, germanium or diamond) are directly responsible for the current measured between the electrodes. The (reverse) bias voltage applied has the purpose of depleting the entire region of the diode, making the detection volume larger and removing intrinsic charge-carriers that would otherwise dominate the induced primary electron-hole pairs. This also makes it less likely for the created charges to recombine and thereby be lost.

  The signal amplitudes from semiconductor detectors are too small to be handled without pre-amplifiers close to the detector.

- The signal in a **wire chamber** is created as the passing charged particle ionises atoms of the gas filling the detector. The separated electrons and ions are accelerated towards the anode and cathode wires, respectively, by an electric field. When the field is strong, then the particles will gain sufficient energy between collisions such that secondary ionisation takes place, multiplying the current. The major part of the current amplification is performed in the vicinity of the read-out wires — by using very thin wires, the electric field strength becomes orders of magnitude larger around them, causing a considerable avalanche the last few 100 μm.

For more information, the reference work [22][2] is suggested.

[2] Although the chapter on digitisation is somewhat antiquated, spending much effort on the use of single-channel analysers.

3.1.1 Measurements: time and energy (loss)

Each signal from a detector is quantified by its time of arrival (relative to other signals), t, and amplitude (on an absolute scale). The amplitude is usually proportional to the energy lost by the particle in passing through the detector, ΔE. Other quantities, most notably the location of a hit in a detector, x and y, are often derived from the above-mentioned quantities during analysis. For segmented detectors, positions are directly inferred from the index of the hit section.

The only thing the DAQ needs to do is to record these two values (t and e) for each signal around each accepted event. Simple and painfully straight-forward.

The only other thing being recorded is the values of a few free-running (i.e. not dead-time affected) counters counting the number of raw and accepted triggers. This is done so that cross sections can be determined even though not all events (that actually happen) are being fully recorded, due to either being minimum bias events (particle enters the setup but nothing happens in the target) or the DAQ being in dead-time (still processing an earlier event).

3.1.2 Reconstruction: time, position and energy (loss)

Let us as an example take a scintillator paddle — long slab, with read-out at the two short ends, see Figure 3.1(left).

Based on the assumption of uniform light propagation in a paddle, one may, given the times and amplitudes of the light measured at the both ends calculate the time (T), position (P, measured from the middle of the paddle) and energy loss (E) of the particle hit in the paddle, see Figure 3.2.

From time measurement

With the effective speed of light in the scintillator, v (usually about $0.5c$ – $0.6c$), the two times measured at the ends of the paddle will be

$$t_1 = T + \frac{1}{v} d_1, \quad \text{and} \tag{3.1a}$$

$$t_2 = T + \frac{1}{v} d_2. \tag{3.1b}$$

From this, we solve for the time of the hit as an average

$$T = \frac{t_1 + t_2}{2} - \frac{d_1 + d_2}{2v} = \frac{t_1 + t_2}{2} - \frac{L}{2v}. \tag{3.2}$$

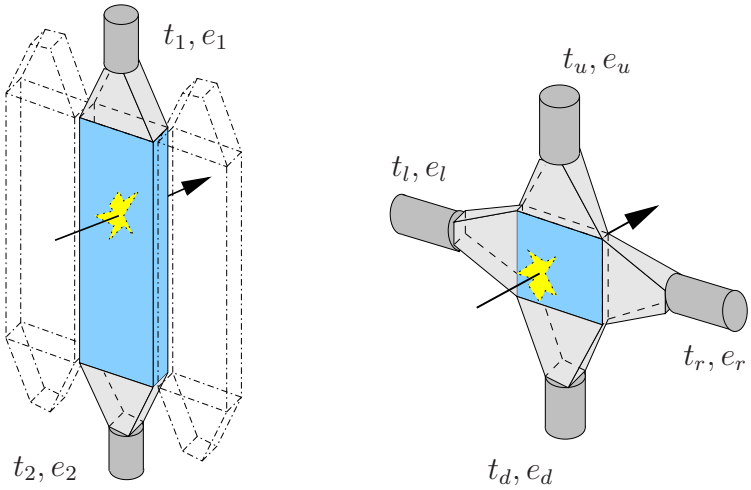

Figure 3.1: Scintillator paddles with PM read-out. To the left a typical TFW paddle. (The LAND paddles are similar, with 10 layers of scintillator, sandwiched with iron converter.) To the right a thin scintillating sheet, used for detecting the incoming beam (POS detector).

The constant length of the paddle, $L = d_1 + d_2$, divided by the effective speed of light, may be fused into the overall time calibration parameters, i.e. just a shift of the time offset for the paddle, leaving us with the more convenient

$$T = \frac{t_1 + t_2}{2}. \tag{3.3}$$

The position is calculated as the difference of the two times, and the paddle length disappears also here from the final formula

$$P = \frac{\left(d_2 - \frac{L}{2}\right) + \left(\frac{L}{2} - d_1\right)}{2} = \frac{d_2 - d_1}{2} = \frac{v(t_1 - T) - v(t_2 - T)}{2} = v\frac{t_1 - t_2}{2}. \tag{3.4}$$

From amplitude measurement

The light attenuation in the paddle can be approximated as an exponential function of the distance between the scintillation location and the PM tube, particularly in cases where the dimension of the paddle along this direction is larger than the other dimensions. When this is not the case, e.g. for the incoming beam detectors (see Figure 3.1, right), quite large abberations may

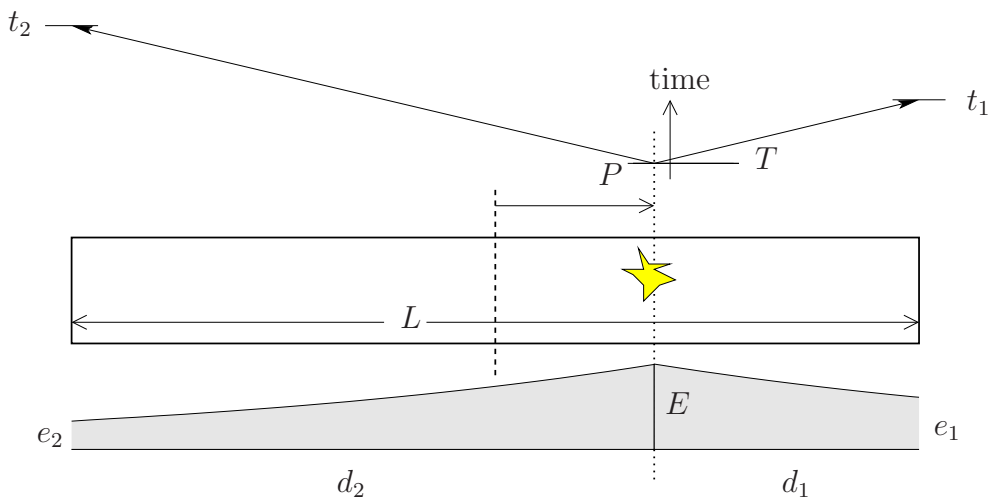

Figure 3.2: Light propagation in a paddle. The time of arrival of the signal (the photons) at the short ends of the paddle depend on the time of the hit and the distance to the hit. The light output also depends on the energy deposited in the paddle, and the attenuation.

occur, i.e. the formulae below need position dependent corrections, usually done with a second order polynomial in the spatial coordinates.

With the light attenuation length in a paddle λ (for TFW about 1.6 m, for POS around 3 cm), the two recorded energies are

$$e_1 = E \exp\left(-\frac{d_1}{\lambda}\right), \quad \text{and} \tag{3.5a}$$

$$e_2 = E \exp\left(-\frac{d_2}{\lambda}\right). \tag{3.5b}$$

We solve for

$$E = \sqrt{e_1 e_2 \exp\left(\frac{d_1}{\lambda} + \frac{d_2}{\lambda}\right)} = \exp\frac{L}{2\lambda} \cdot \sqrt{e_1 e_2}, \quad \text{and} \tag{3.6}$$

$$P = \frac{d_2 - d_1}{2} = \frac{\lambda \ln \frac{E}{e_2} - \lambda \ln \frac{E}{e_1}}{2} = \lambda \frac{\ln e_1 - \ln e_2}{2}. \tag{3.7}$$

Once again, a constant factor, $e^{\frac{L}{2\lambda}}$, may be omitted and this time considered part of the energy gain calibration parameters, giving

$$E = \sqrt{e_1 e_2}. \tag{3.8}$$

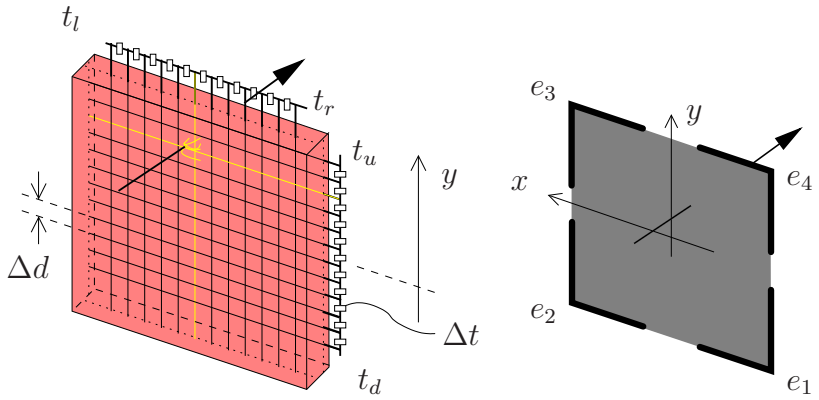

Figure 3.3: Wire chamber with delay-line read-out (left) (ZST). Position sensitive silicon detector (PSP), with signal (charge) division between the electrodes in the corners (right).

With delay-line read-out

The read-out of a wire chamber with delay-lines (to reduce the number of read-out channels to two per coordinate, x and y, see Figure 3.3(left)) gives rise to very similar formulas for the reconstruction of the position from the two times,

$$y = v\frac{t_d - t_u}{2}, \tag{3.9}$$

and similarly for x, where the signal propagation speed now is $v = \frac{\Delta d}{\Delta t}$. Δd is the distance between two wires, and Δt is the propagation delay between them.

Silicon detectors

In the particular experiment this thesis is centred around (S245), only two silicon detectors were used. The purpose of the two semiconductor diodes was the determination of the ion charge before and after the target via energy-loss measurements. With a single read-out channel the reconstruction of the total energy deposited is trivial.

For silicon detectors with read-outs at the corners, see Figure 3.3(right), the deposited energy/charge is shared between the pick-ups (eventually read-out channels) in an additive way. The summed sharing is usually not much affected by location-dependent attenuation. Therefore, for this case, $\ln E$ is of no use.

For such a detector, the deposited energy is then simply

$$E = e_1 + e_2 + e_3 + e_4. \tag{3.10}$$

Assuming a linear dependence of the charge collection on position, the normalised positions (within the natural coordinate interval of the detector, $[-1, 1]$) are reconstructed as

$$P_x = \frac{E_{\text{left}} - E_{\text{right}}}{E} = \frac{(e_2 + e_3) - (e_1 + e_4)}{e_1 + e_2 + e_3 + e_4}, \quad \text{and} \tag{3.11}$$

$$P_y = \frac{E_{\text{up}} - E_{\text{down}}}{E} = \frac{(e_3 + e_4) - (e_1 + e_2)}{e_1 + e_2 + e_3 + e_4}. \tag{3.12}$$

As for the case of amplitudes measured with an exponential position-dependent relation to the energy loss, and the accompanying position reconstruction using logarithms, it is important to have the energies here expressed with a correct absolute zero. For this case it is in addition necessary that the four individual energies all have correct gain factors applied before the formulas above are used.

3.1.3 Tracking the way to momenta

The basic idea of the setup is simple: since one can not observe how a reaction itself happens, measurements are limited to a before-and-after view of the involved nuclei. Using the fact that the reactions here, for reasons of simplified physics interpretation, are performed at high energies, enough detectors[3] can be placed and operated before and after the experimental target such that the incoming and outgoing particles can be fully characterised. This means an event-wise unambiguous determination of the isotope of each ion and its momentum vector. This setup does not determine properties whose event-wise measurements would only be statistical (like polarisation). The exception is the gamma-ray measurement around the target location, which is somewhat "statistical", being subject to the limited efficiency even of 4π gamma detectors.

When the hits in each detector have been reconstructed, they are combined into tracks. The detectors are no longer considered as of how they function, but what variables $(x, y, t, \Delta e)$ they measure in the lab system. The ions, incoming and outgoing, are identified with the characteristics of each track: the measured energy loss in the detectors, the calculated curvature in any magnetic field passed, and the velocity. The velocity and coordinate measurements also give the four-momenta of the particles.

[3]Sufficient energy (velocity) is necessary to allow the particles being detected also to pass through a set of detectors while losing energy.

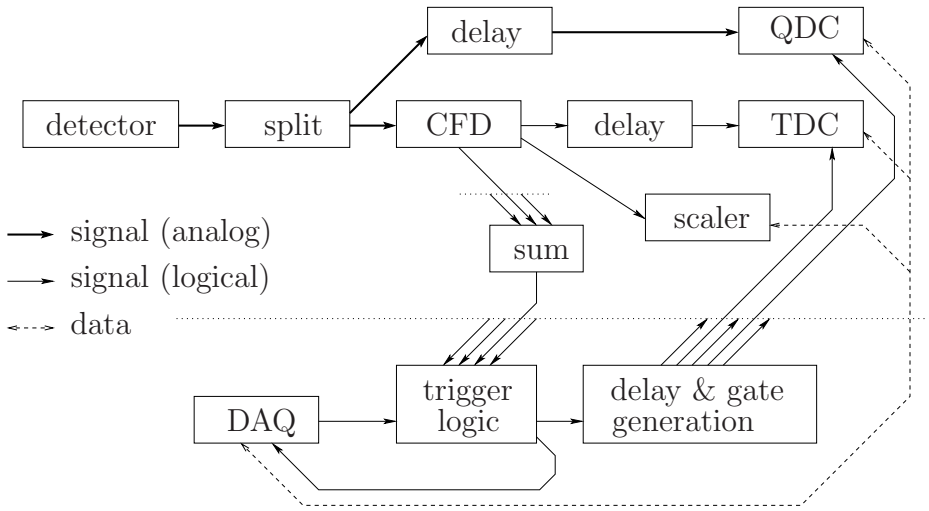

Figure 3.4: Electronics chain for one channel, and its interaction with the trigger electronics and the DAQ program. (See the Glossary and Figure 11.6 therein for more details.)

3.2 Electronics and data acquisition

The virtues and benefits of physics experiments with the kinematically complete ALADiN-LAND reaction setup in Cave B, during this work dismantled and re-ignited in Cave C, have already been briefly covered in the introduction, and extensively elsewhere [5]. Instead, we will take a look from a DAQ perspective.

The electronics for the LAND setup was until 2006 divided into two parts. The detectors with many channels (segmented detectors located behind the target) have their full read-out (split-card, QDC, CFD, delay, TDC, and scaler) in the cave not too far away from the detectors. Figure 3.4 is a schematic representative of almost any detector channel. All signals from the detectors with few channels (mostly upstream the target) are transported outside the cave to the Messhütte[4] where their full electronics is housed. The only exception is a few preamplifiers that must be close to their Si-detectors at the beam line. The trigger logic is also housed in the Messhütte. The electronics in the cave provides trigger signals (sum/or) from the large detectors to the trigger electronics, which in return delivers the master start to all read-out electronics (including the cave).

[4]Counting room with or without electronics.

The DAQ/digitisation itself consist of one FASTBUS crate[5] in the cave surrounded by about 10 CAMAC crates of constant fraction discriminators, delay modules and scalers. In the Messhütte, the DAQ controls two CAMAC crates, one mainly housing TDCs and QDCs (and an ADC for some silicon pin diodes), the other one is used for a pattern unit and a few scalers recording the actual triggers, and two logic matrices controlling the trigger conditions. It also has a programmable time calibrator module, used to gain-match all TDCs in the setup. Some 15 NIM crates are also used.

Further subsystems were added to the setup in 2007. Two detector system types, silicon micro-strip detectors and drift-chambers for proton detection, with many channels and custom electronics, proved to be reliable sources of DAQ operator headache. An additional VME crate with a few standard TDCs and QDCs was also added to handle a new time-of-flight wall for fragments.

The data acquisition is running the Multi Branch System, a general DAQ framework actively developed at GSI [10]. It supports the use of several read-out processors taking data, being synchronised via trigger modules interconnected with a trigger bus (cable), cf. Section 7.1. The MBS system handles all data transport and file writing after the data has been put into the event buffers in the processor. The read-out functions, "talking" to the actual hardware, are implemented as user functions (as they are experiment specific) and then compiled and linked into one of the programs making up the full MBS system. The user function is not only responsible for the event-wise read-out, but also for initialising the DAMs at startup and to control other periodic tasks, like reacting to special time calibrator events by doing the programming and firing of the time calibrator module. It also assures a reliable and responsible handling of the A/QDC zero-suppression by continuously measuring the pedestals, see Section 3.3.3 of [1].

3.3 The S245 experiment

The S245 experiment was carried out in Cave B at GSI in September 2001. It is the third in a row of studies of light neutron-rich halo nuclei in complete kinematics at the ALADiN-LAND setup, following S034 [23, 24, 25] and S135 [26, 27, 28, 29]. Isotopes were produced via projectile fragmentation by letting a beam of ^{18}O from the SIS impinge on a thick production target of

[5]Since the move of the setup to Cave C, there are actually two FASTBUS crates — since the first one has gotten old and cranky, and is hardly able to work reliably when it is completely filled with modules. It is also mechanically much easier to handle two half-full crates, as with only one the cables to the modules become very densely packed.

Be (4.007 g/cm^2) and selecting the exotic nuclei of interest (^{8}He, ^{11}Li and ^{14}Be) by their magnetic rigidity when passing the FRS [30]. By using two different primary beam energies, the composition of the radioactive beam could be varied, see Table 3.1. The energy was chosen as high as possible to maximise the LAND efficiency [31] and the angular coverage of the detectors by the forward-focusing of charged fragments and neutrons, but limited by the bending power of ALADiN — unreacted beam and charged fragments must be bent away enough not to hit the neutron-detector LAND.

Table 3.1: Energies and intensities [32] of the exotic beam in the S245 experiment. The selected magnetic rigidty was fixed at 9.52 Tm in both settings, giving the same radioactive beam energies for each of the exotic isotopes, regardless of the primary beam energy.

	Setting	$E_{\mathrm{kin}}(^{18}\mathrm{O})$ (MeV/A)	^{8}He	^{11}Li	^{14}Be
E_{kin} (MeV/A)			240	280	305
Intensity (s^{-1})	^{8}He	308	365	73	0
Intensity (s^{-1})	^{14}Be	360.5	2–3	45	36

The previous experiments have examined the halo phenomenon both using C and Pb targets, probing mainly via the strong as well as the electromagnetic interaction, respectively. This time a liquid hydrogen target of 350 mg/cm^2 thickness was employed, kept in a cylindrical tube with walls of 100 μm Mylar, 50 mm long and with a diameter of 30 mm. Measurements for background subtraction were made with gaseous hydrogen filling the container. The small charge of the target ensures the reactions to be almost exclusively of nuclear origin, i.e. mediated by the strong force. As the target is a nucleon, and not a nucleus, no internal degrees of freedom can be excited (at the energies available in the experiment).

One exciting feature is the possibility of detection of scattered protons from the target, as their characterisation would complete the kinematics picture of each event. The limited angular coverage of the proton branch, especially the fraction of the azimuthal angle φ available around the target perpendicular to the beam, restricted however the amount of such events. This has been addressed in later experiments (on other nuclei) by using silicon micro-strip detectors covering all φ and the enclosing gamma-ray detector for determination of energy of the recoil proton [33]. See Figure 3.5 for more details on the experimental setup. A longer description of the S245 experimental setup is in Paper V, and a detailed account of its analysis can

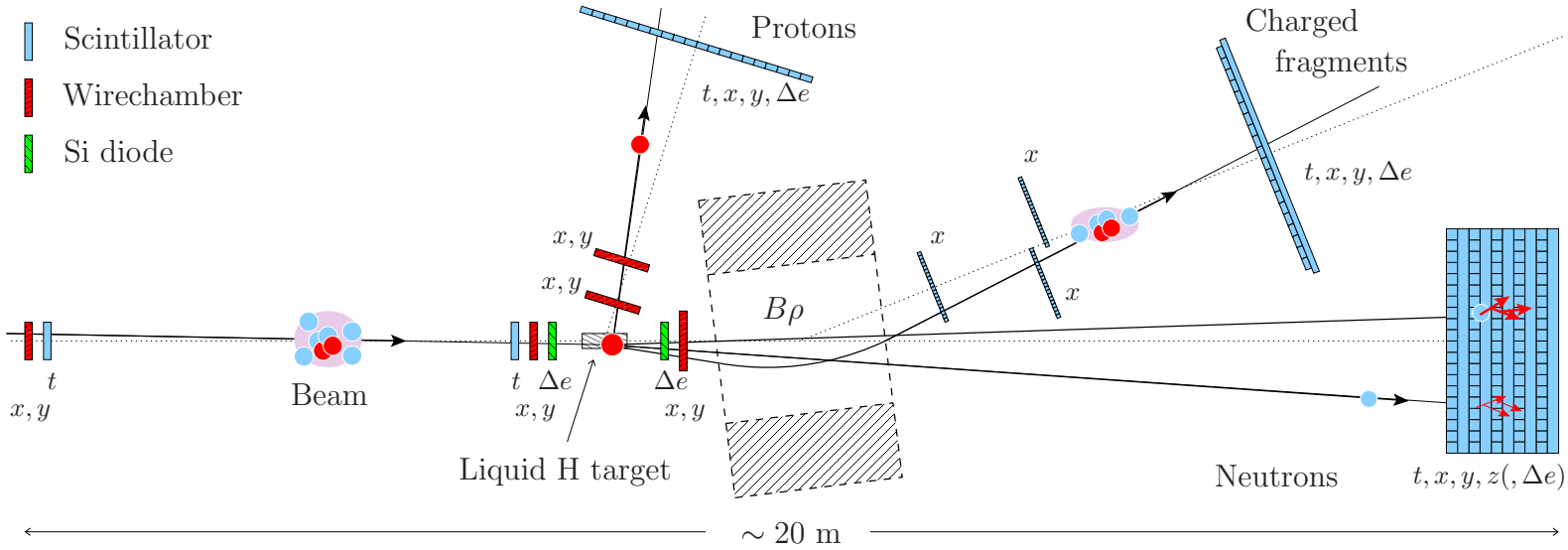

Figure 3.5: The S245 experimental setup (September 2001, Cave B, GSI), not to scale. The incoming beam is identified by means of time-of-flight between two scintillators and energy loss in a silicon detector. The (x,y) position on the target is determined with a MWPC. Outgoing fragments have their angle measured by a MWPC and the velocity by the time-of-flight to the TFW. The isotope is determined by means of energy loss in a silicon diode (charge) and the track curvature in the magnetic field of ALADiN (mass). Neutrons are detected in LAND giving both the velocities (with a TOF measurement) and the outgoing angles. Target-like protons are detected at angles between $66\,^\circ - 81\,^\circ$ in the lab: their direction with two MWPCs, and energy and ID via the punch-through pattern in the TOF detector. Also see Figure 10.1.

be found in [5].

The remainder of this thesis (after the next Chapter) will only implicitly touch the particular subject of the S245 experiment, as it deals with the use and development of a complete analysis system for the LAND setup, `land02`, re-usable also for future experiments[6]. Starting to work with the analysis, it became clear that some problems have their roots in the DAQ...

[6]The author likes to believe that it is the insistence on the use of solutions, not hacks, that has made the software fit for use for half a dozen further experiments performed in 2005 and later. S245 itself has gained a lot from the code-level collaboration thus possible.

Chapter 4

Physics outcome

It would give the chronicler, whose devotion to his Art is equalled only by his distaste for work, considerable pleasure to discourse at some length on the overpowering silence which invaded the cave and the invisible reactions which took place in it — besides bringing him several pages nearer to the conclusion of this fourth chapter. The fusillade of works which a student of S245 may despairingly liken to "a display of fancy shooting in which all the ions are beautifully halo'ed around the edge of the target, away from its centre" tugs almost irresistibly at his trigger finger, see Figure 4.1. The sudden removal of neutrons from ^{11}Li, the precise knock-out of a proton from the heavy core of ^{14}Be, the tightening of the thin straight section of the beampipe, the clenching of the vacuum flanges and the involuntary upward lift of the target container — all these and many other manifestations of *e*motion could be the subject of an essay in descriptive prose in which the historian could wallow happily for at least a thousand words.[1]

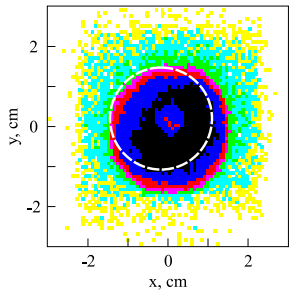

Figure 4.1: The halo beam profile of halo isotopes in S245. The dashed line shows the analysis cut, derived from the deduced [34] target position.

4.1 Structure beyond the neutron dripline

As previously mentioned, there are structural peculiarities of nuclei at the driplines. We are interested in investigating these effects also beyond the driplines. Such unbound systems are so far only reachable experimentally for

[1] Resistance was futile — this is a shameless rip-off of a tense scene in [35], adapted for a nuclear setting.

Table 4.1: Experiments described in the included Papers. The reactions are described by the notation: target(beam,detected)undetected.

Paper	System	Facility/Exp	Beam	Target	Reaction
I	^{10}Li	REX-ISOLDE	^{9}Li	d (C_3D_6)	d(^{9}Li,p)^{10}Li
II	^{10}Li	GSI/S245	^{11}Li	^{1}H	p(^{11}Li,n^{9}Li)pn
	^{12}Li		^{14}Be	^{1}H	p(^{14}Be,n^{11}Li)ppn
	^{13}Li		^{14}Be	^{1}H	p(^{14}Be,nn^{11}Li)pp
III	^{7}He	KVI	d	^{7}Li	^{7}Li(d,pp)^{7}He
IV	^{7}He	GSI/S245	^{8}He	^{1}H	p(^{8}He,n^{6}He)pn
V	^{9}He	GSI/S245	^{11}Li	^{1}H	p(^{11}Li,n^{8}He)ppn
	^{10}He		^{11}Li	^{1}H	p(^{11}Li,nn^{8}He)pp

the lightest nuclei, through demanding experiments due to small statistics and difficulties achieving good resolutions. Moreover, the states (resonances etc.) are broad, mixing with each other. However, we can use different probes to access the information, ultimately observing the same features employing different boundary conditions, see Table 4.1.

4.1.1 ^{10}Li in transfer and knock-out experiments

The unbound ^{10}Li system, one of the two-body subsystems of the Borromean ^{11}Li nucleus, has been the subject of many studies, see Table I of Paper I. The results of an experiment approaching ^{10}Li via the transfer reaction d(^{9}Li,p)^{10}Li at REX-ISOLDE, where the outgoing proton is detected, is reported in Paper I. An s-wave virtual scattering state with a negative scattering length a around -13 fm to -24 fm and a p-wave resonance at ~ 0.38 MeV with a width of ~ 0.2 MeV were deduced from the data.

The first part of Paper II is concerned with the reaction channel of the S245 experiment where one neutron was knocked out from ^{11}Li, leaving ^{10}Li. The unbound system promptly fell apart into ^{9}Li+n, with both residues detected: the charged fragment in the TFW after passing through the ALADiN magnet and the neutron in LAND. The invariant mass method again revealed (see Figure 4.2): an s-wave scattering state with $a = -22.4(4.8)$ fm and a p-wave resonance at 0.566(14) MeV with a width of 0.548(30) MeV.

4.1.2 The systems ^{12}Li and ^{13}Li

Following event-by-event selection of ^{14}Be incoming ions and ^{11}Li in the outgoing channel with the simultaneous detection of one or two neutrons, the

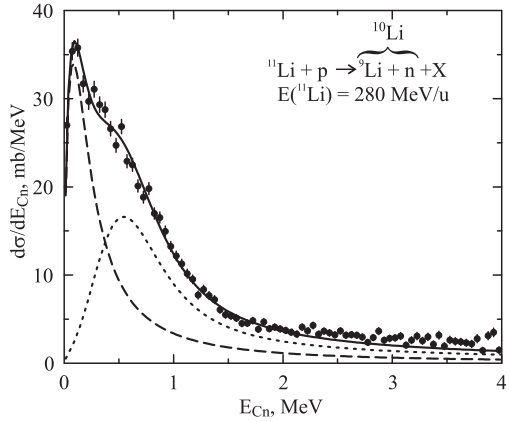

Figure 4.2: Differential cross section as function of the ^{9}Li+n relative energy after one-neutron knock-out from ^{11}Li. The dashed line represents a virtual *s*-state and the dotted line a *p*-wave resonance in ^{10}Li. From [5], also in Paper II.

unbound systems ^{12}Li and ^{13}Li have been observed for the first time, as reported in Paper II.

The relative-energy spectrum of ^{12}Li was reconstructed from the measured momenta of its decay products, ^{11}Li+n (Figure 4.3 (left)), and can be described as a virtual s-state with a scattering length of $-13.7(1.6)$ fm. No further low-lying resonances were found. In this reaction channel, one neutron was also removed from ^{14}Be and not detected in LAND, presumably with a momentum large enough to be effectively separated from the ^{12}Li final state.

When observing ^{11}Li+n+n after break-up of ^{14}Be, it is very likely that the observed distribution (Figure 4.3 (right)) is strongly influenced by the structure of ^{14}Be itself, in addition to any resonances in ^{13}Li, manifested by enhancing the observed cross-section at some relative energies of the ^{11}Li+n+n system. Knowledge of the initial state is therefore necessary to be able to get a handle of features of the final states. Calculations of the expected correlated background from ^{14}Be show that the influence of the initial state is dominating the invariant-mass spectrum, but cannot completely account for the strength at low energies, which therefore is attributed to the final-state interactions of ^{13}Li as a resonance at 1.47(31) MeV.

It is quite remarkable that it, in a large fraction of the events with incoming weakly-bound neutron-rich nuclei, is possible to knock one strongly bound proton out of the halo core without completely shattering it such that

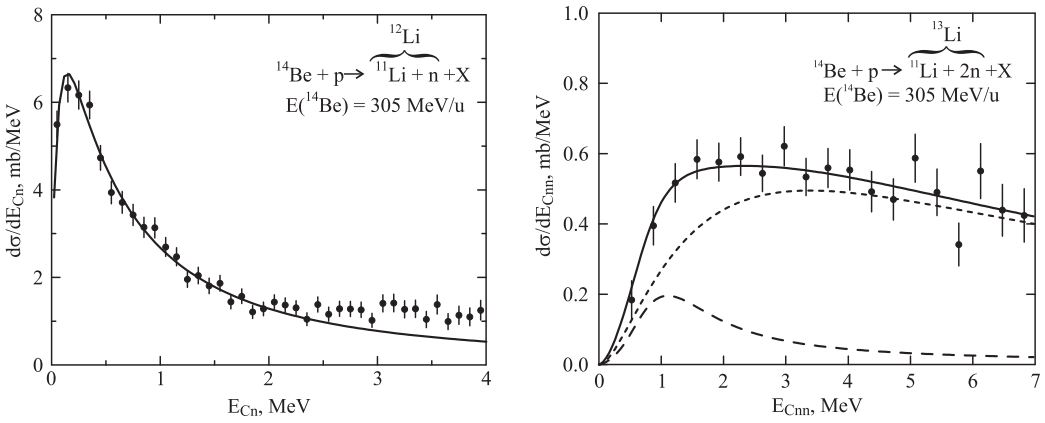

Figure 4.3: Differential cross section as function of the ^{11}Li+n (left) and ^{11}Li+n+n (right) relative energy after proton knock-out from ^{14}Be. The unbound ^{12}Li (left) is described by a virtual *s*-state. The dotted line in the right picture corresponds to correlated background from ^{14}Be, while the dashed line represents a resonance in ^{13}Li. From [5], also in Paper II.

the remaining core and halo neutrons are still able to form a new halo system. Even though the newly formed systems in this case are unbound and naturally disintegrate rapidly, it is possible to do their spectroscopy via the decay products.

4.1.3 The systems ^{9}He and ^{10}He

The data set provided by the S245 experiment with a mixed incoming beam is very rich. In Paper V the proton knock-out channel from ^{11}Li is employed to reconstruct heavy, unbound He systems. The systems are studied in a manner similar to the unbound Li systems, now with the detection of ^{8}He and one or two neutrons, representing possible final states of ^{9}He and ^{10}He.

Both these systems have been studied before, see Tables 1 and 3 of Paper V. The earlier experiments on ^{9}He are mainly double-charge-exchange [36, 37, 38] through multi-nucleon transfer or pion double-charge-exchange reactions [39, 40], all creating ^{9}He from ^{9}Be. Another experiment has studied isobaric analog states in ^{9}Li, i.e. configurations similar to those in ^{9}He, by the reaction p(^{8}He,p)^{8}He [41]. ^{9}He has also been approached in a manner akin to the REX-ISOLDE studies of ^{10}Li above, but at higher energies, in the reaction d(^{8}He,p)^{9}He [42], detecting the proton. Except for the elastic resonance scattering search for isobaric analog states, all these experiments retrieved

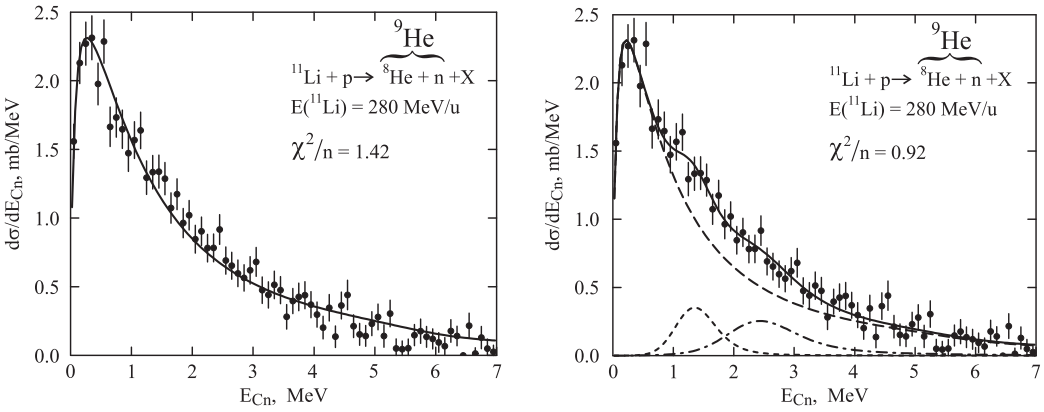

Figure 4.4: Differential cross section as function of the ^{8}He+n relative energy after proton knock-out from ^{11}Li. The description of ^{9}He by a single virtual s-state in the left picture is rejected with 98.6 % confidence due to the bad $\chi^2/n = 1.42$ of the fit. The description to the right includes two additional (dash-dotted and dotted) resonances on top of a (dashed) s-state. From [5], also in Paper V.

information on ^{9}He by detection of the other remaining (ejectile) system after the reaction.

In our experiment, we reconstruct the energy of the ^{9}He final state from the relative energy of its disintegration residues, ^{8}He+n, see Figure 4.4. The relative-energy spectrum is found to be dominated by an s-wave with a scattering length $a = -3.17(66)$ fm. This alone does not produce a satisfactory fit; by guidance of previous experiments [38, 40] two additional resonances are included, one at 1.33(8) MeV and one at 2.4 MeV. The available statistics does not allow all parameters of the fit to be simultaneously varied, forcing us to make use of the earlier data to fix the values for the two widths and the location of the latter resonance. This procedure yields a fit that agrees well with the data, thus confirming the findings of previous experiments.

The existing data on ^{10}He are not as consistent, see the discussions in Paper V. The data from the present experiment, based on the measurement of ^{8}He+n+n, see Figure 4.5, itself allows for multiple interpretations. As a starting point, the spectrum can not be solely described by a correlated background stemming from ^{11}Li, but can then be reproduced in two ways. Either as the correlated background and one resonance at 1.42(10) MeV with a width of 1.04(76) MeV. Alternatively, the entire spectrum can be fitted as two resonances at 1.54(11) MeV and 3.99(26) MeV. Since the second in-

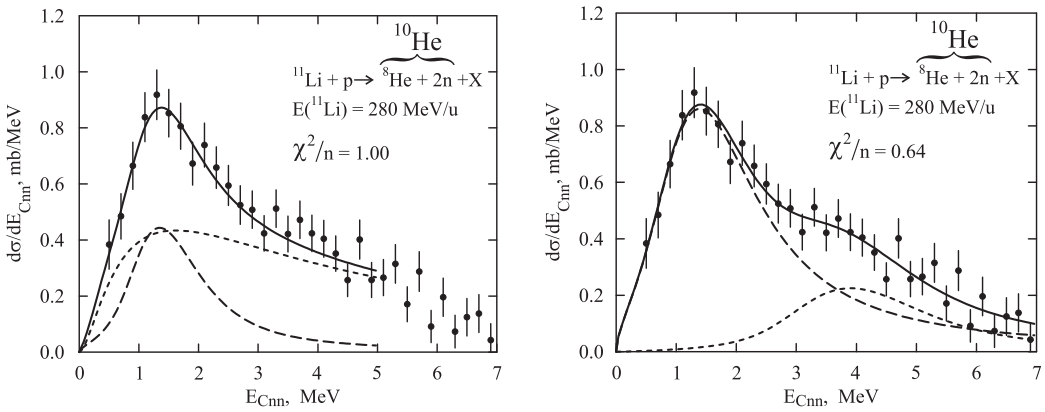

Figure 4.5: Differential cross section as function of the ^{8}He+n+n relative energy after proton knock-out from ^{11}Li. The left picture describes ^{10}He as a correlated background from ^{11}Li and one resonance. The same data is described in the right picture as two resonances. No preference is given. From [5], also in Paper V.

terpretation does away with the correlated background, it implies a reaction where there is no influence of the ^{11}Li structure through the final-state interaction of ^{10}He. This duality of interpretations may be resolved by studies of energy and angular correlations between the fragments [43], used successfully with data from earlier halo experiments at the LAND setup [44, 29]. Angular correlations can give insights about the structure of a system, not available in relative-energy spectra, and help define quantum numbers of observed states.

Correlation plots for ^{8}He+n+n are shown in Figure 4.6 (preliminary data, not presented in the Paper). They use the differential cross-sections with respect to internal variables describing the cluster system in two different coordinate systems (see inset plots of Figure 4.6 to the right and left). The differential cross-sections are projected onto the partial energies of and angles between the Jacobi momenta. The partial energy ε is the energy in the "directly connected" subsystem (n+n to the left and ^{8}He+n to the right in Figure 4.6) divided by the energy of the whole system. The second variable is the angle θ between the respective Jacobi momenta $\vec{\mathcal{P}}_1$ and $\vec{\mathcal{P}}_2$. As the projections describe the same system, the four curves must be fitted simultaneously, constraining each other. In this case, the preliminary correlation study suggests an interpretation with two resonances, based on the mismatch with the expectation for correlations stemming from ^{11}Li.

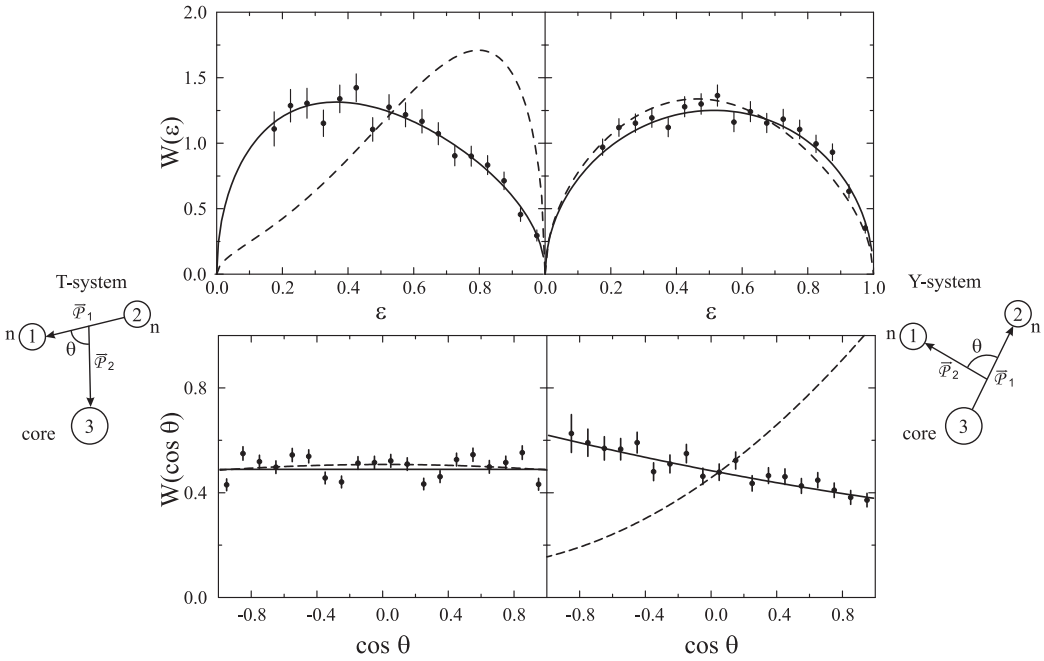

Figure 4.6: Energy (upper) and angular (lower) correlations in the system ^{8}He+n+n, with variables describing it in two different coordinate systems. Events are selected with ^{8}He+n+n relative energy in the region of the ground state, $0 < E_{\mathrm{Cnn}} < 3$ MeV. Results of the fit are shown by a solid line. The dashed line represents correlations obtained from the wave function of ^{11}Li assuming no final state interaction. Preliminary, from [5].

4.2 Weakening of the spin-orbit coupling at the driplines

The unbound nucleus ^{7}He was studied in the previous light halo nuclei experiment S135 [28] at the ALADiN-LAND setup, where among others, ^{8}He was allowed to break up on carbon and lead targets. The ground state resonance of ^{7}He is expected to have $I^{\pi} = 3/2^{-}$ [45], corresponding to a $p_{3/2}$ valence neutron, in line with the Shell Model (SM). As the obtained relative-energy spectrum of ^{6}He+n (Figure 4.7 (left)) could not be explained by the $p_{3/2}$ resonance alone, one more low-lying resonance was claimed. The origin of this second state was interpreted as being the $p_{1/2}$ spin-orbit partner of the $p_{3/2}$ ground state. The small separation of these two states implies a weakening of the spin-orbit coupling compared to what is observed in nuclei closer to the

valley of stability. It is a manifestation of the same phenomenon that gives rise to "new magic numbers" by shifting and also ultimately reordering levels of the SM. One example of level-inversion is ^{11}Be, whose ground state is $1/2^+$ (with a $s_{1/2}$ valence neutron) although it would be expected to be $1/2^-$ (with a $p_{3/2}$ valence neutron). Another example is the recent interest attracted by ^{24}O, showing signs of being a double magic nucleus [46]. The "conventional" expectation for being doubly magic would be ^{28}O with $Z = 8$ and $N = 20$, but if the $d_{3/2}$ levels are shifted up, the larger energy gap associated with a magic number instead appears already for $N = 16$.

The finding in ^{7}He has led to a lot of renewed experimental activity, to verify and further investigate its level-structure, where some have found a state at 1 MeV or lower and others not. One may perhaps say that the "plot" has thickened — but not always due to statistics?

4.2.1 ^{7}He investigated in a charge-exchange reaction

Paper III reports on an experiment performed at KVI where a deuteron beam is used to deliver a neutron to ^{7}Li, replacing it for an outgoing proton, thereby creating ^{7}He. Both this and the proton remaining from the deuteron are detected in coincidence, together described with the moniker ^{2}He, giving a reaction description ^{7}Li(d,^{2}He)^{7}He (or ^{7}Li(d,pp)^{7}He). The reconstruction of the two protons allows the energy of the remaining system ^{7}He to be inferred. While nicely resolving its ground state, and other break-up mechanisms (e.g. ^{4}He+t), the purported $p_{1/2}$ state could not be observed. The data suggests a possible resonance at an approximate excitation energy of 1.45 MeV with a width of 2.0 MeV when the quasifree charge-exchange process on ^{7}Li is constrained by a measurement of the ^{6}Li(d,^{2}He)^{6}He reaction, which was investigated at the same setup in another experiment [47].

4.2.2 ^{7}He from knock-out

Break-up of ^{8}He was also studied in S245, now on a proton target. The spectrum of ^{7}He as ^{6}He+n (Figure 4.7 (right)) is analysed and compared with the results of [28] in Paper IV. The obtained distribution could not accommodate a low-lying $p_{1/2}$ resonance. A target-dependent re-scattering was suggested as one possible cause of the deviation from the shape prescribed by a pure $p_{3/2}$ state in the spectrum of [28]. Paper IV also includes a re-analysis of the spectrum of ^{5}He [27] obtained in S135 to get a reliable estimate of single-particle reduced width, which could then be applied to ^{7}He of the current experiment via mass scaling. The spectroscopic factor for the ^{6}He(0^+)+n configuration in ^{7}He could thus be determined to be $S_n = 0.61(3)$, being

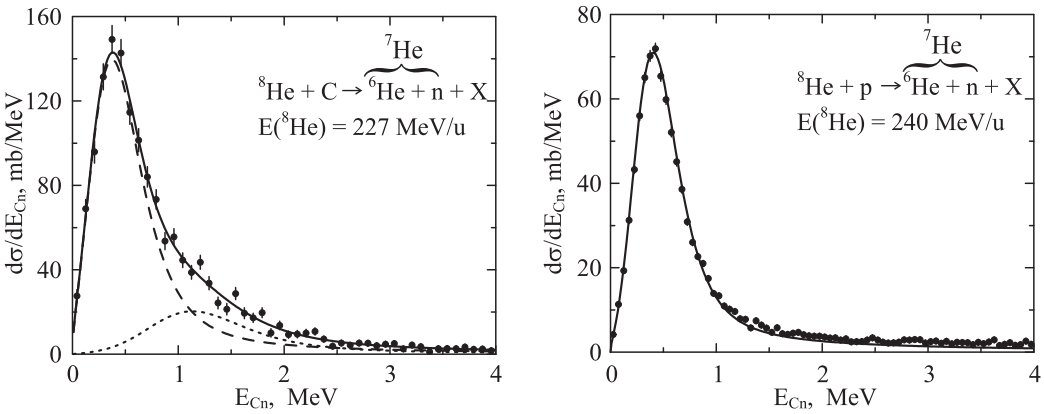

Figure 4.7: Differential cross section as function of the ^{6}He+n relative energy after one-neutron known-out from ^{8}He on a carbon target (data from [28], S135, left) and a liquid hydrogen target (S245, right). Both experiments observe the ^{7}He ground state resonance, while the spectrum of ^{6}He+n after ^{8}He break-up on carbon has additional strength, interpreted as an additional resonance [28]. Pictures courtesy of [32], also in Paper IV.

different from unity, it shows that the ground state of ^{7}He is not a pure single-particle $p_{3/2}$-state.

4.2.3 Further work

The issue on the location of the $p_{1/2}$ state in ^{7}He has not been conclusively resolved. What more can be done with the ALADiN-LAND setup to shed more light on the situation?

The break-up reaction on the proton target involves two outgoing particles that are not measured, the recoil proton of the target, and one neutron. The neutron is scattered to large angles and not detected by LAND, which in itself is the tag for the identification of ^{7}He as ^{6}He+n, and not just being one of the components in ^{6}He+n+n. Detection of the recoil proton would give an additional handle on the reaction mechanism and allow further selection of events to be made. One example would be to select events closely compatible with quasi-free scattering of the proton on the (undetected) knocked out neutron. When that neutron has come in (with ^{8}He) as a loosely bound halo neutron, it would essentially behave as a neutron beam and in cases of quasi-free scattering on a target proton, the kinematics of the recoil proton is completely determined. One may even imagine that such a selection could en-

hance some currently hidden (swamped/rare) component of the bare ^{6}He+n spectrum. Some limited experimental means for such an event selection were present in S245, see Section 3.3.

One goal of the development of the `land02` analysis tool is to make it cheap — less labour-intense — to revisit old data sets, performing the full cycle of calibrations and reconstruction again. Such a job was recently carried out [48] with the deuteron break-up run from 1992 whose data is still used to normalise and simulate the LAND efficiency. It would be interesting to also re-process the S135 data[2], if only to affirm the results[3].

[2]The author would estimate that such an undertaking may become suitable for a diploma work in about a year. The main missing pieces are code to reconstruct data from the WDC charged fragments detectors (see Appendix A) which have not been used in later experiments, a planned overhaul of the calibration parameter handling system, and the generic tracker simplifying ion identification.

[3] *"In the good old days physicists repeated each other's experiments, just to be sure. Today they stick to FORTRAN, so that they can share each other's programs, bugs included."* Edsger W. Dijkstra [49]

Chapter 5

First stage of analysis – reading raw data files

The data collected in nuclear and particle physics experiments is just numbers, arranged as events — each describing a random outcome of the ions gambling the high stakes of Quantum Mechanics. Analysing an experiment is to make a sum greater than the parts.

5.1 Anatomy of a raw data file

Being the needle's eye through which the experiment preparations and execution reaches the analysis and results, well organised, structured and robust raw data files are needed. Their layout naturally resembles that of multi-level Matryoshka dolls — let us crack the composition of data files open from inside out[1]:

5.1.1 Events

Most present day state-of-the-art nuclear physics experiments are performed by recording what happens to individual ions in many independent reactions, each called an event. All data related to a particular event is stored together, and usually has not much to do with any previous or following event. (Or at least, only as little with the previous one as the hundred ones before that).

The decision to store the information about a particular event is taken by the trigger of the DAQ, and as the data is digitised and collected at the same time, it also becomes natural to store it together.

[1] Think of a chicken picking its way out, rather than blowing a valuable piece of toy up.

5.1.2 Payload – the actual data

The data from each DAM channel in an event is a digital number, representing a time interval or a pulse amplitude or integral. It is common for electronics to have and deliver 12 bits worth of resolution. Either all channels from a module may be read out, in which case no indexing information is needed, or zero-suppression is employed, such that each channel delivering information in an event must be accompanied by its designation — the channel number within the module. This information is combined into one data-word per channel during data read-out, and comprises the actual payload and innermost component of the raw data files. In figure 5.1 it is marked `ch, value`.

5.1.3 Headers – index meta-data

Modules

To complete the identification of a channel, also the module must be given. The module's location (e.g. slot number) is often given in a header word which precedes the data of that module. The number of data words following, i.e. from that module, can also be given in the header word(s). The count may be used to provide a check of the data contents, as well as an easy way to skip over the data, for any readers not interested in that piece of data. The module headers are symbolised by `slot, words` in the figure. When the data length is given in a trailing module footer word (which may be necessary, depending on how the module works internally), only checking is possible.

Sub-events

With several crates of DAMs participating in an experiment, it is convenient to arrange the data

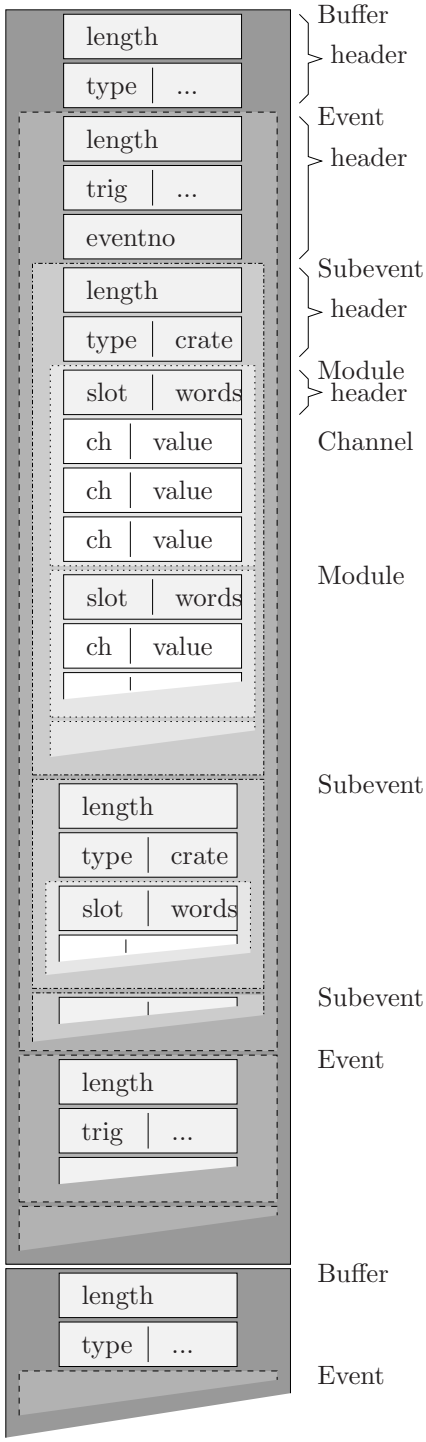

Figure 5.1: Raw data file anatomy.

from each crate in a separate part of the event. There are several reasons:

- Each crate usually deals with one or a few specific detector (types) of the setup, making the separation natural.
- Each crate may be read out by its own processor, with the data merged into events only at a later stage, requiring a unified wrapping of the elements (fragments) until that point.
- Analysis, particularly calibrations, may only be interesting in specific parts of the setup, which is simplified by the unified packaging, allowing easy skipping of unused elements.

This is achieved by the sub-event headers, marked length, `type` and `crate` in the figure.

Events

The sub-events of an event are finally combined and written to the file following an event header, giving the total length, an event number and some other information. These are denoted `length, trig, ..., eventno` in the figure. E.g. a trigger number can denote the class of an event — physics, calibration, periodic pulser, etc.

5.1.4 Buffers

While the variable-sized events in principle can be written directly to a file, one after another, another level of packaging is usually employed: buffers. The buffers are fixed-size (or semi-fixed size, modulo a convenient size) chunks of event data, with a header. The headers, marked `length, type`, can contain overall information (like coarse time stamps, checksums, etc.). These are either not necessary or too expensive to store per event.

The major drawback of fixed length buffers is that events too long to fit at the end of a buffer either must be postponed to the next buffer, wasting valuable space, or be split between the current and one (or several following) buffers, causing grief during unpacking.

The buffers are by some considered a relic of the times when data was recorded directly to magnetic tape, which worked with fixed size records. Even on disk-based storage, the buffers still provide some rigidity to the data files, e.g. making recovery in case of damaged files more straight-forward. Having, even variable-sized, buffers in the file format, allowing for markers with less granularity than events has further advantages. It can be used to

mark the presence of e.g. slow control events (see Section 8.1.1) and allow for simpler verification and skipping. Raw data files from event-based experiments are naturally of a streaming nature. Using any kind of index or other contraption at the beginning or end of the file is not a good solution, as it prevents easy streaming of data through pipes.

Implementation considerations

Abolishing the use of fixed-length records does not necessarily make it easier to construct a high-performance unpacker. An efficient unpacker can in any case not allow itself to do a `read` system call per event, and moreover also needs to avoid any copying of the input data. The preferred concept is to use a `mmap`-approach. Any unaligned `read` guarantees a copy to happen. The alignment is versus the system page size, usually 4 kiB. Therefore, unless variable-sized or non-existent buffers are handled with great care, the fragmentation issue will rear its ugly head during unpacking in any case.

In the author's opinion, fixed-size buffers may well be surrendered — if only to make it easier when someone wants to quickly write a simple unpacker, without the need to consider fragmentation. Having buffers encapsulating events is however too useful to be abandoned — the alternative of sprinkling the auxiliary information of the buffer headers between the events makes its presence random and can therefore not be counted upon.

Section 2.3 of Write-up U has a summary of the data packaging formats handled by UCESB. It also discusses the mayhems of endianess issues and unwarranted byte-swapping.

5.2 The unpack problem

Unpacking is the operation of translating the raw data produced by a (running) DAQ system, into a format usable by some more generic analysis system, either this is a more user-friendly graphical display system, or batch processing jobs crunching the physics out of the data.[2]

[2]Unpacking and mapping is sometimes called 'data sorting', perhaps also including the following analysis stages of applying calibration parameters. As that name at least to the author implies or suggests the use of $O(n \ln n)$ sorting algorithms in the data mapping stage, while $O(n)$ bucket-sort strategies are more adequate, the term 'data sorting' is strongly disfavoured.

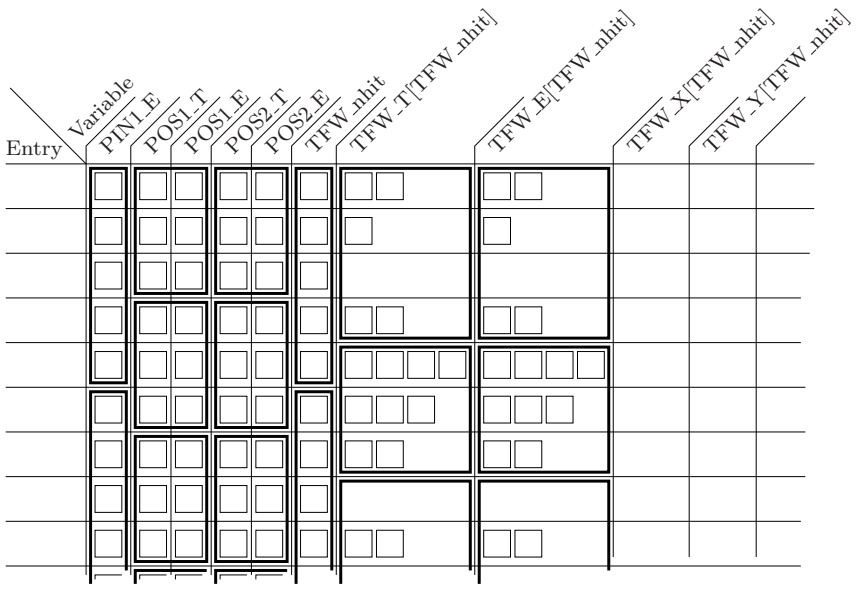

Figure 5.2: HBOOK column-wise ntuple files and ROOT trees are stored with individual variables or groups in buckets, each containing data from many events. This allows analysis (e.g. plotting) to operate on only a few of the available items per entry while accessing the storage media in large continuous chunks. The figure is not to scale — the number of events in each bucket is much larger in reality.

5.2.1 Ntuples – PAW and ROOT

The information from the experimental setup can be viewed as a large matrix. Each column describes a particular variable, e.g. a raw value of a detector channel or something more elaborate, like an ion's charge or mass. Each row in the matrix describes one event. This matrix is often quite sparse, as most detector channels are only fired in a small fraction of the events. With a few hundred million rows (events) and some thousand columns (channels and derived values), such a direct storage scheme is neither practical nor feasible.

The data is naturally generated row-by-row and efficiently stored as described in the previous Section. The presentation of data (to the physicist user) with e.g. the generic analysis tools PAW [50] or ROOT [51] is however almost exclusively done column-wise — with histograms and scatter-plots of (functions of) one, two or a few variables, possibly with cuts and selections on some other.

5.2.2 Why not skip the raw data stage?

As the files with data arranged column-wise obviously are wanted for the later, higher-level analysis, why not directly store the information recorded by the DAQ in such a file? There are two reasons not to skip the raw data stage:

Unpacking solves a real problem. The payload of the raw data files is usually kept as close as possible to the data produced by the electronic modules (DAMs). The modules (and their data format) are carefully crafted to give precise[3] and compact (using zero-suppression and bit-flags) data. This, often ancillary, information can be very valuable when figuring out and dealing with potential malfunctions and mishaps during the experiment (often found out months or years later). But handling them to some benefit can be quite laborious and also usually would not fit into the higher-level analysis schemes: the major problem is then not always the lack of data, but also the copious amounts of extra information.

While the above mentioned analysis tools are very efficient in handling and plotting large sets of (mostly floating point) data, the pictures quickly become overcrowded when needing to deal with all the 'extra dimensions' that many extra flags can make. In some sense, it is the job of the unpacking (and very early analysis stages) to consume and take into account the ancillary information and by that also making it superfluous, not having to propagate it. One example is a detector-independent tracker, see Chapter 10. It has no way of dealing with the particularities of any particular detectors, and neither can nor will take any but the kind of information it has requested into account.

Hiding a job doesn't make it go away. The data recorded for each channel of a detector is often comprised of values from several modules, e.g. a time and an amplitude. Those are recorded by different modules. During analysis, this information should be kept together. During read-out, it is however delivered ordered by module. As a result, mapping of the data must occur after recording but before analysis.

As every line of program code is a potential source of errors (bugs), especially the DAQ programs must be kept simple. The reason is that everything that is corrupted in the recorded data files, cannot be corrected and must be laboriously counter-acted in analysis programs every time the files are

[3]Not only in terms of resolution, but any particular conditions that apply... (overflows, underflows, event counters...)

processed. An unpacker (which essentially is what is needed for a DAQ to write the data as column-wise files) is a complicated piece of software, and can act as an infinite source of persistent bugs and problems in the recorded data if operated as a part of the DAQ before storing the raw data. When errors are discovered in an analysis program, one simply corrects the program and runs it again. As mapping can be done while processing recorded files, it should therefore be done on recorded files. Another reason is that the mapping information itself may be incorrect during data-taking[4].

Another reason to keep the data in the raw files arranged and unmodified, as delivered by the modules, is the operation of the DAQ itself. Problems often occur module-wise, and debugging is simplified by having access to the pristine data.

[4]Which seems to be the common case, experience shows, even with the best of intentions...

Chapter 6

Detectors – DAQ and analysis

With an experimental arrangement as the ALADiN-LAND setup, built from several quite independent detectors, the calibration and reconstruction work is most easily handled separately. This makes calibration boot-strapping of each device possible, irrespective of what other detectors are available in any particular experiment. Tailor-made routines are needed to leverage the strengths of each detector. As several stages of the reconstruction (and calibration) behave similarly for different devices, it is however often possible to combine efforts.

The handling of the TFW and LAND detectors was already covered in Chapter 4 of [1]. The overall layout of the analysis process with different data levels was thereby also described. It is akin to the one used by `ucesb` (Write-up U), as the latter was modelled after the former. This makes both design concepts descendants of the previously used analysis software [52]. This Chapter describes handling of data from the GFI detectors, outlining Paper VI, but also touching on a few possibilities not covered there.

6.1 GFI calibration and reconstruction

The large area scintillating fibre detectors (GFI: Großer Fiber Detektor), have been in use at the LAND setup for more than ten years [53], but only recently (within this work, see Paper VI) have they gotten an appropriate software companion. Today, their calibration and reconstruction is handled with much less effort, steps towards multi-hit recognition have been taken, and scarce manpower resources can be directed towards new goals.

The GFI detectors provide a good example of a detector where calibration and reconstruction are well separated, joined by the interface of "orthogonal" and well-defined calibration parameters. But they are also devices where

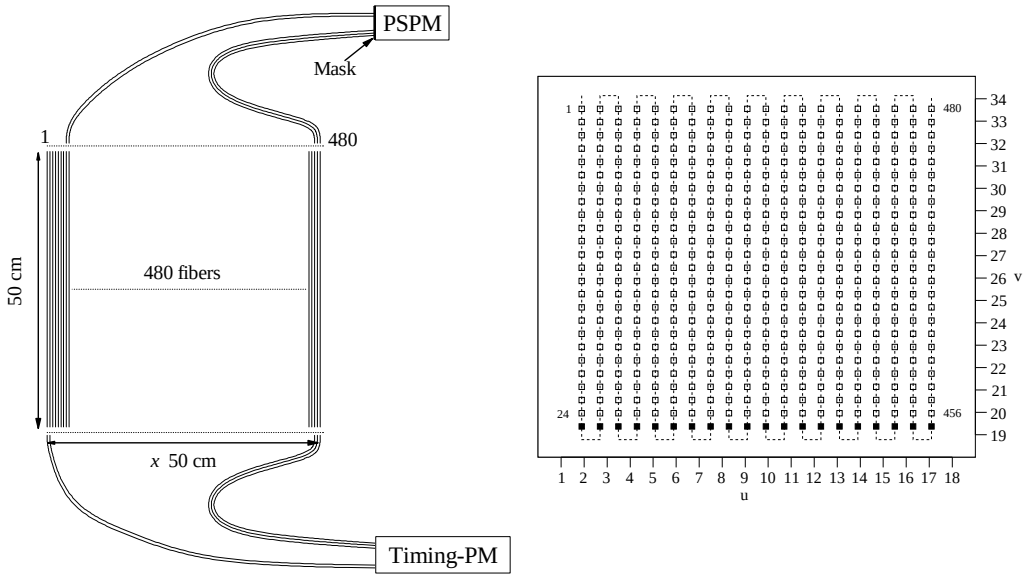

Figure 6.1: Each GFI detector has approximately 480 fibres, each 1 mm wide. One end of each fibre is connected to a normal PM tube and the other to a position sensitive PM tube (PSPM) to determine the hit location. The right picture shows the mask used to align the fibres on the PSPM.

calibration and reconstruction must run in lock-step, such that the determination of the second layer of parameters requires the existence of the first layer of parameters and reconstruction.

6.1.1 The GFI detector hardware

The GFI detectors comprise a 50×50 cm^2 quadratic sensitive area of 1 mm wide vertical scintillating fibres, held with a C-shaped support structure [53], see Figure 6.1 (left). The purpose of the detectors is to make measurements of the horizontal position along the tracks of charged fragments behind the ALADiN magnet, see Figure 3.5.

The scintillation light from each end of each fibre is transported by optical light guides of equal length to two PM tubes. The light guides from one side are connected together to a single non-position sensitive PM tube, from which timing information can be extracted, e.g. for trigger purposes. The light guides from the other ends are arranged using a mask and coupled onto a position sensitive PM tube (PSPM). With the mask, see Figure 6.1 (right), the around 500 fibres are placed to give signals in a regular pattern on the

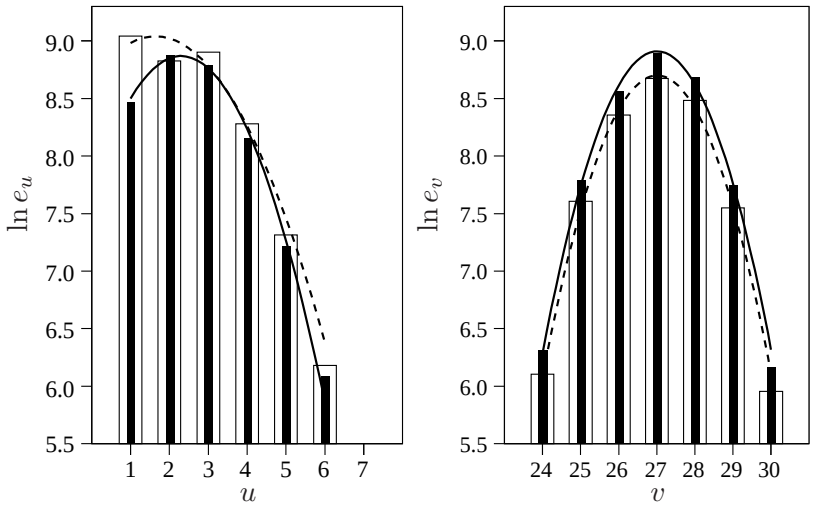

Figure 6.2: The light from a hit scintillating fibre causes signals (shown in logarithmic scale) in several neighbouring wires in both the u and v anode planes. The open bars and dashed line represent raw amplitude values and fit, corrected after gain adjustment to the filled bars and solid line.

PM tube front face.

The amplification in the PSPM of the electrons originating from the photons is performed using a layered wire mesh, such that the signals read out from the back side of the PM tube give an image, although broadened, of the location of the fibre in the mask that delivered the photons. Each hit results in a nearly Gaussian-shaped distribution, with a handful of wires delivering signals, see Figure 6.2. The read-out on the back-side is done with wires in two orthogonal directions, each having 18 and 16 wires, respectively. These 34 signals are then further amplified before individual digitisation by a QDC.

Under circumstances where the energy deposit, and thus the final signal-to-noise ratio is large enough, such that individual fibres can be reconstructed, one has managed to read 500 individual fibres with only a fraction of the number of electronics channels. In cases where the signals are smaller, due to lighter, lower charged ions[1], the requirements on the position resolution are fortunately also lower. As the ions are lighter, the relative mass differences are larger, and the requirements on the reconstruction of the deflection in the magnetic field of ALADiN are correspondingly reduced.

[1]Bethe formula, $\Delta E/dx \sim Z^2$, see Eq. (1.1).

The fibres are arranged in a meander scheme in the mask, with a smaller spacing between neighbouring fibres in each line and a larger distance between the lines. This reduces the risk of inter-line misreconstructions at an increased chance of reconstructing a neighbour fibre as hit instead. But a neighbour error is only 1 mm, while an inter-line error is anything between 1 mm and 5 cm for detectors where the fibres are arranged in an up-down-up pattern as in Figure 6.1, and a constant 2.5 cm for an up-up-up arrangement. This makes the detector resolution become on average 2.5 cm instead of the optimal 1 mm for the lightest ions where the capability to distinguish fibres is lost.

6.1.2 New GFI analysis software

As a part of the analysis of the S245 experiment (light halo nuclei), an effort to re-implement the tools to handle the GFI data, using more formalised ideas, was initiated. The goal was, and still is, to be able to use them for detection of multiple outgoing particles, e.g. two He ions from the break-up of ^{14}Be. The project was completed together with Kripamay Mahata — who did most of the work — mostly using data from the later S287 experiment. Using data from an experiment with heavier ions, such as Ni in S287, had the advantage of delivering much clearer signals, making development easier. The code has been used for online calibrations in several recent experiments, as well as being tested with older data.

The routines have been tested for S245, although they have not been used for production there yet, i.e. the results of the included Papers that concern S245 have been obtained without using data from the GFIs. The reason for this is that in an experiment with light ions, where the efficiency of the detectors is far from one, the set of detectors delivering data in each event will vary a lot. This complicates track reconstruction considerably — to be solved by a general tracker, as described in Chapter 10.

Reconstruction

The reconstruction proceeds in several steps, each reversing one stage in the process of data collection, from digitised values to ions passing through the detector:

- The values from all channels are “equalised” (normalised), such that they have the same relative strength. This corrects for varying amplifier gains before the QDC and differing signal collection efficiencies in the PM tube.

This is performed by generic code in the reconstruction program, as read-out channels representing amplitudes of basically every detector need this treatment. The output of this stage is represented at the `sync` data level of the analysis program, see [1], Section 4.5.

- The centres, u_0 and v_0, (and the heights, $A_{u,v}$) of the hits in the two orthogonal wire planes, called u and v, are separately determined. As the approximate fit function is a Gaussian,

$$e_{u,i} = A_u \cdot e^{-\frac{(u_i - u_0)^2}{d_u^2}} \quad \text{and} \quad e_{v,j} = A_v \cdot e^{-\frac{(v_j - v_0)^2}{d_v^2}}, \tag{6.1}$$

  one may instead fit a parabola to the logarithms of the amplitudes,

$$\ln e_{u,i} = e_{0,u} - \frac{(u_i - u_0)^2}{d^2} \quad \text{and} \quad \ln e_{v,j} = e_{0,v} - \frac{(v_j - v_0)^2}{d_v^2}, \tag{6.2}$$

  where $e_{0,u,v}$ represents the amplitude. Fitting a parabola has the advantage of the parameters being directly calculable using analytical expressions, as opposed to being obtained through a time consuming minimisation of the residual function. To avoid severe effects on the fit by small values (at the edges of the parabola), whose uncertainties get magnified by the logarithms, all values are weighted with the inverse of the original amplitudes. This corresponds to treating the uncertainties as coming from the statistics of photon counting.

  To further limit the fit, only 3 channels on each side of the channel with the highest value are included. Depending on the relative locations on the mask, the detectors are in some circumstances able to reconstruct multiple hits in an event. This is performed by doing another peak search after excluding the wires included in the first parabola and subtracting the Gaussian function from the previous fit from the remaining wires.

  The coordinate of the hit, (u_0, v_0), makes up the `dhit` data level of the detector.

- The PSPM hit coordinate can now be compared to the known locations of the mask of fibres. Internally, each fibre location is associated with an integer index (k, l) giving a picture of the mask. In order to make the look-up quickly, the possible range of k and l indices are precalculated for each unit square of the (u, v) space.

  Each, usually deformed, quadrilateral within the possible k, l range, mapped on the (u, v) space, is now tested to see if it contains the hit

coordinate, in which case also a fractional location within the quadrilateral in k, l coordinates can be calculated. The original idea was that these fractional indices could be usefully mapped onto the x coordinates along with uncertainties of the hit belonging to a particular fibre. This turned out to unnecessarily complicate things, and only the information on which square is associated with the calculated k_0, l_0 is used. (The code for reverse look-up of deformed quadrilaterals found use in the PSP detector reconstruction instead.)

- The k_0, l_0 indices found are used to select a range of points on the mask, $k_0 - 1 < k < k_0 + 1, l_0 - 1 < l < l_0 + 1$ for which a weighted average of their respective fibre positions x is calculated. The weights are calculated as from a Gaussian distribution of the distance of the hit (u_0, v_0) to each nominal fibre position $(u_\mathrm{f}, v_\mathrm{f})$, divided by the sum of the uncertainty of the hit position plus the calibrated width/uncertainty of the fibre position. The standard deviation of the summed x distribution is used as the output uncertainty at the detector `hit` level, together with the nominal $1/\sqrt{12}$ mm for one fibre[2].

  When the position uncertainties are small (generally much smaller than the mask inter-fibre distances) the above method will single out only one fibre. However, when the hit has been reconstructed in-between the nominal positions of fibres, the weighting exponentials decrease so quickly that, also now, often only one fibre is singled out by the average, which is not wanted. To make those cases behave more realistically, also a small but broad exponential component is added, dominating only for in-between hits that thus become associated with larger errors instead of being artificially pulled towards one or another closest fibre.

With this, the detector specific reconstruction is finished, and the obtained x coordinate together with estimated accuracy is passed along to the general tracker, see Chapter 10. Or possibly several coordinates, when handling multiple hits.

Other quantities than horizontal position

Even if the x position is the quantity the GFI detectors are designed to primarily deliver, extraction of other parameters, based on the signal amplitudes and the measured time, should also be considered.

The amplitudes obtained from the parabolic fits in the u and v coordinates may be regarded as good PSPM-internal variables (after gain-matching).

[2]The standard deviation of a square function of length a is $\frac{a}{\sqrt{12}}$.

They can be used to match the multiple peaks in u and v of the same event together. But they are not necessarily immediately representative of the energy loss of an ion, due to varying efficiency of the optical coupling between each fibre and the PM tubes.

In the following, only single hit events are considered, as the normal timing PM tube at the other end is always involved, and it has no ability to disentangle double hits.

When combining the amplitude information from both PM tubes, at both ends of the scintillating fibre, both the energy loss and vertical position can be reconstructed, using the principles in Section 3.1.2, i.e. Eqs. (3.7) and (3.8). Due to the need to correct for the individual light coupling efficiency at the PM tubes, only events which can be uniquely assigned to a certain fibre can be handled. Another issue is that the fibres are fairly thin, only 1 mm wide, making it quite likely for ions, especially traversing the detector at a horizontal angle, to cut the edge of the fibre and only go through a fraction of the thickness. The registered energy loss is then diminished accordingly. The ion charges calculated from the energy losses must therefore be regarded with scepticism, while the positions reconstructed would be correct, as only the relative amount of light recorded at the two ends matters. The same problem applies to all detectors where the interaction volumes are discretely segmented, e.g. the TFW.

As only one time is available, the vertical position, y, cannot be reconstructed from a time difference measured between the top and bottom PM tubes. The one time can be used to give the time of ion passage though, but as the scintillating fibres are 50 cm long, an uncertainty of about 1.5 ns is caused just by the effective speed of light in the fibre around 16 cm/ns. This can be corrected for by using the y position as determined from the signal amplitudes, or if that is unavailable, in a more roundabout way:

After the tracking routine has determined what tracks can be made out of the hits from various detectors, it will also be able to calculate the track parameters at all detectors, in this case the y coordinate of the intersection at the GFI detector. (This would usually be the same thing as y interpolated between the target and the TFW, the time-of-flight wall located further downstream, with intrinsic capability of determining also spatial hit coordinates.) The estimated vertical position can now be used to correct the measured time, and reduce the associated error.

Calibration parameters

The reconstruction described above is straightforward to perform for each event, given the existence of a number of input parameters:

- Gain factors. To normalise the amplitudes of the read-out channels relative to each other.

- Locations $(u_\mathrm{f}, v_\mathrm{f})$ of the fibres, and their widths $(\sigma_{u_\mathrm{f}}, \sigma_{v_\mathrm{f}})$. This information is for practical purposes arranged in an integer k, l grid.

- The fibre ordering on the mask, associating each k, l grid point to an actual fibre x location on the detector. These are determined by the glued order of the fibres and only hardware instance dependent, unchanged between experiments.

- To retrieve more information than just the x coordinate, further parameters are needed: individual fibre light coupling efficiencies, the light attenuation length, the effective speed of light in the scintillator and also walk correction for the timing channel.

Calibration

Determining a set of calibration parameters is done by running the reconstruction routines up to the point where the parameters are needed. Then a routine similar to the intended reconstruction is run but where the output variable for each event is an estimate of the desired calibration parameter. By collecting estimates from many events a good average value is determined, generally by a fit, disregarding outliers as noise. Events can often be selected such that some special condition is known to be fulfilled, e.g. fixing the normal output variable in the reconstruction, making it possible to solve for the calibration parameter instead. In other cases, the normal reconstruction routine is used, with the calibration parameters temporarily set to a nominal value (e.g. 0 for additive or 1 for multiplicative parameters), and the actual values are determined with some optimisation strategy.

Most satisfactory are the cases when the calibration parameters can be directly calculated. Less desirable is when one has to resort to cumbersome iterative procedures.

- The relative gain parameters of the signal amplitudes from the PSPM wires are determined by an iterative procedure. As each hit causes a charge distribution over several wires, the single spectra of each wire would contain all values between 0 and a maximum. The maximum is not constant, as some wires are located directly below a row of fibres in the mask, while others are in-between rows — further accentuated by mask distortions that occur when e.g. the PSPM is close enough to the ALADiN field to affect the trajectories of the electrons in the

PM amplification stages. In addition, the wires at the edges of the PSPM never correspond to the maximum charges, as the fibre pattern is smaller (to allow for complete Gaussians at the fibre mask edges).

There is also no a priori way to select particular events (e.g. with maximum at a certain wire) — without the reconstruction / fitting itself.

Instead, it is assumed that the gain is approximately equalised from the outset, only needing smaller corrections. For each event the parabola fitting is performed, and information is collected so as to adjust all amplitudes simultaneously, to:

- Improve the fit qualities, i.e. reduce the deviation of individual wires from the fitted parabolas.
- Make the height of the fitted parabolas in the u and v coordinates match for each event.

The necessary corrections are calculated all together by solving an overdetermined equation system.

The second item above has the advantage of not only making u and v use a common reference frame, but also connects wires at opposite ends in one coordinate via two equations going through the orthogonal coordinate. Using only the parabola deviations they would be loosely connected via multiple "hops" in the equation system.

The gain matching improves the fibre mask pattern remarkably. Several "ghost" fibre points, as well as deformed patterns at the edges are corrected. This procedure is made possible by the fitting of the collected charge distributions, as it allows the deviation of single wires to be estimated. The previous reconstruction software, which was content to determine the (u, v) positions with simple centre-of-gravity calculations, had no possibility to assess its own quality.

- When the gain parameters are optimised, it is relatively straightforward to determine the locations of the fibres on the PSPM mask, see Paper VI.

The parameters needed to use the ancillary information from the amplitudes and or timing are examples of parameters that can be directly calculated by solving for them and thus running an "inverted" form of the calibration. Their determination do however require external references, i.e. information from other detectors or fitted tracks.

6.2 Outlook: detector-internal calibrations with Millepede II

Millepede II [54], is a wonderful — almost magical — set of computer programs that can solve a set of equations which regularly pop up as a part of performing detector calibrations [55].

An over-determined detector system[3] that, in each event has multiple correlated hits or signals, can use these correlations to extract calibration parameters. This is also what was done in the previous sections, but with quite some organisatorical pain due to the fact that the parameter determination is a chicken-and-egg problem. To determine the parameters using the correlations, a rough reconstruction — which requires the parameters — must be performed. This can be seen in Eq. (6.1), rewritten to give the amplitude (e_i) in one GFI anode wire (w_i) as a function of the fibre location (w_0), energy loss (E), and the wire gain parameter (g_i):

$$e_i = g_i \cdot E \cdot e^{-(w_i - w_0)^2/2d^2}, \tag{6.3}$$

or, as simplifies the calculations, in logarithmic scale:

$$\ln e_i = \ln g_i + \ln E - (w_i - w_0)^2/2d^2. \tag{6.4}$$

For a set of e_i in an event, the local parameters E, w_0 and d can be determined (fitted), given the global gain parameters g_i. This cyclic dependency was broken in Section 6.1.2 by initially assuming the gain parameters to be fixed (e.g. unity). The gain parameters were then successively approximated as the average deviations in each wire from (6.4).

Millepede II provides the means to solve for the sought gain parameters in (6.4), as if the local parameters — for all involved events — would be solved for simultaneously. The critical point is *as if*. With some hundred thousand events included in the calibration, there are several hundred thousand local parameters involved. But only the few ten global gain parameters are of interest. (Once the gain parameters have been successfully determined, the local parameters — which are interesting per event — can be calculated during normal event reconstruction.)

The Millepede II system does this with modest requirements and appealing features [54]:

- Each equation is described by the difference of the left- and right-hand side of e.g. (6.4) and the derivatives of all appearing local and global

[3] Either several detectors, or several segments within a detector.

parameters. The user is therefore not required to solve any equation, only describe how a measured value corresponds to an event, and be able to differentiate that.

When the equations describing the hits are non-linear, it is however still necessary to obtain approximations for the non-linear parameters.

- The information is collected by the `mille` part of the system, which is easily included and used by the experiment-specific event processing programs. It generates a file in a simple format with the information of all equations.
- In the best UNIX tradition, the actual equation solving is carried out by the stand-alone program `pede`. This alleviates compilation and linking issues.
- Built-in out-lier removal. `pede` is able to automatically exclude "noise" events in the solution process.

6.2.1 Gain matching revisited

The gain matching part of the GFI calibration serves as a good example and playground for employing Millepede II to detector calibrations tasks. The derivatives of the right-hand side of Eq. (6.4) with respect to the local (E, w_0 and d) and global (g_i) parameters are needed; $1/2d^2$ is for simplicity replaced by c and the logarithms are conveniently "forgotten" (made part of e_i, E_i and g_i):

$$\frac{\partial e_i}{\partial E} = 1, \tag{6.5a}$$

$$\frac{\partial e_i}{\partial w_0} = 2c(w_i - w_0), \tag{6.5b}$$

$$\frac{\partial e_i}{\partial c} = -(w_i - w_0)^2, \tag{6.5c}$$

$$\frac{\partial e_i}{\partial g_i} = 1. \tag{6.5d}$$

This has been tested and results in calibration parameters similar to those determined by the iterative method described earlier. Some discrepancies appear for the outer-most wires, which are slightly ill-determined as they only sample the tails of the Gaussian distributions. Due to the non-linearities, which require approximate values for w_0 and c to be determined for each event, the method has problems when the start-approximations for the gain

parameters are too far off. The routine described earlier handles this better, and is used in a first iteration step to obtain reasonable start-approximations.

It is foreseen to use Millepede II for the detector alignment needed by the generic tracker described in Chapter 10. It is also expected that it can be used to determine parameters necessary to correct for effects that crop up when e.g. the linear Eqs. (3.6) and (3.7) provide a too simple description of a detector element. The advantage of the aforementioned equations is that they allow for direct solutions of the calibration parameters, as described in [1]. However, effects due to non-linear corrections have recently caused problems in the analysis of the NTF. Also here, the procedure would be to use the existing routines to determine start-approximations for the dominating linear parameters, and use Millepede II for fine-tuning.

Chapter 7

MBS improvements

The Multi Branch System (MBS) [10] is a DAQ framework developed and maintained at GSI. It is used at over 100 installations world-wide. The job of the MBS is to set up memory mappings to access the various data acquisition modules (DAMs) and to handle trigger synchronisation between multiple sub-systems. Furthermore, it takes care of transport and combination of the sub-system data into complete events. At a first glance, the most prominent weakness seems to be that the MBS does not handle the actual DAM setup or read-out. This however turns out to be its greatest strength! Not limiting the user to a predefined set of modules and read-out modes, but instead leveraging the full power of a user-written read-out function per subsystem in the `C` programming language makes it extensible and flexible. This is especially important when combining a multitude of separate detectors — developed more or less independently — into larger setups.

This chapter begins with an overview of the MBS data transport layers. It is followed by the description of a few modifications developed and tested during a few recent experiments in 2007 at the LAND setup, to prevent their suffering from scaling problems of the network data transport encountered as the number of sub-systems (processor read-out nodes) increased from two to six. The improvements include ways to ensure a unique event numbering in multi-event-builder setups as well as a few performance enhancements.

The overview also forms the background of the next chapter, which contains a proposal for how much needed 'persistent' (slow control) events can be incorporated into the MBS data transport stack.

Figure 7.1 outlines the components of a single-crate MBS system, with data flowing from the DAMs to permanent storage as well as being available for on-line analysis.

7.1 Data propagation

Each crate with a processor participating in a MBS set has a trigger module. All trigger modules are connected with a trigger bus. The trigger modules handle the dead-time locking and the master trigger module[1] is responsible for accepting triggers. For every accepted trigger, each trigger module will notify its processor with an interrupt that an event is pending for read-out[2]. When done, the read-out program, `m_read_meb`[3], will notify the trigger module to release the local dead-time. When all systems have completed their read-out, the global dead-time is released and another trigger can be accepted. The MBS will ensure that the data from all subsystems of each accepted trigger eventually ends up as one complete event in the data file.

On each read-out node, the data for each subevent, as formatted by the user readout function, `f_user_readout`, compiled into the `m_read_meb` program, is passed in a subevent pipe to the `m_collector` process. If the processor running the `m_collector` process can access the memory of all readout pipes[4], it can combine the subevents of each event into complete events, which are packaged into fixed-size buffers. These buffers are then consumed by the `m_transport` process, which can write them to tape, disk or via RFIO as well as serve them to a network client. In addition, the `m_stream_serv` or `m_event_serv` programmes can also be used to deliver formatted buffers to network clients for on-line analysis.

7.1.1 Subevent pipes

A subevent pipe is a shared memory segment (possibly even sitting in another VME module). It is used to pass *sub*events (also called fragments) from the readout (`m_read_meb`) to the collector (`m_collector` or `m_ds`), or from one collector to another collector (running on another processor, via shared memory). The (final) collector or the receiving `m_dr` is responsible for formatting the (several) fragments of each subsystem into a full event. This is also where the event number is inserted. (In the pipes, events only have

[1] In each MBS system, one trigger module is designated as master. When several systems, having separate dead-time domains, are loosely coupled together with time-stamps, each domain has its own master. The capability to handle such an additional system combination layer is not dealt with further in this write-up; its mechanics are straight-forward.

[2] It is also possible to run the system such that e.g. only each nth trigger causes an interrupt, provided the DAMs have multi-event buffering capabilities.

[3] Programs making up the MBS system generally have the prefix `m_`, and `f_` denote functions within the source.

[4] Direct access to the subevent pipes is possible if it is a single system, or for systems connected with a common databus, e.g. VSB or VME.

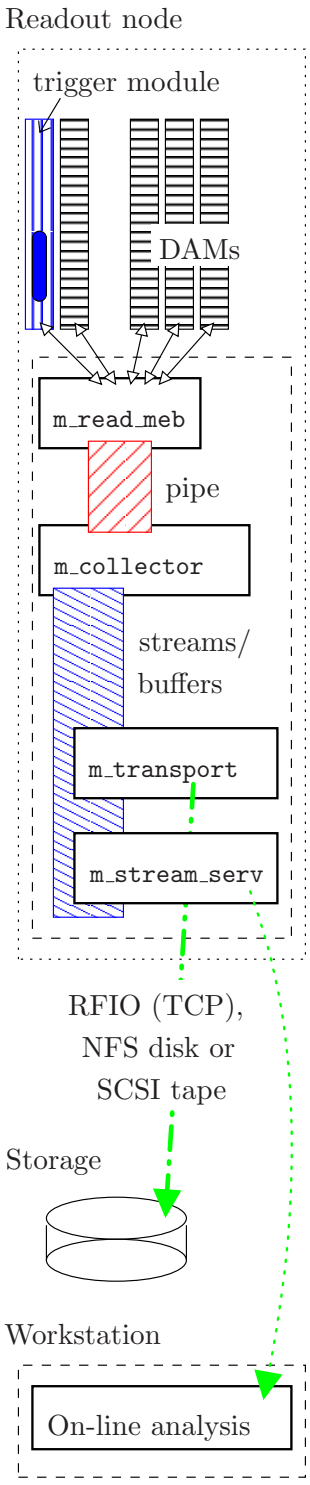

Figure 7.1: Single-crate MBS data flow.

the five-bit counters from the trigger modules for synchronisation checking.)

A pipe consists of two parts, see Figure 7.2. The first piece of memory is a circular buffer of `PIPE_LEN` control structures with meta-data (the trigger type, 5 bit counter, length and pointer into the data field) for each fragment. The remaining memory is used to store the actual data. It is into this memory space that the `pl_dat` destination pointer of `f_user_readout` points.

This data space is split into two, to make the locking (to ensure that old, yet uncollected events are not overwritten) between the readout and collector easy. This split has the unfortunate side-effect when using delayed event building, that in practice only half the memory can be used until forced event building kicks in. As long as there are free meta-data slots available and the current half-pipe has enough space left, or the other half-pipe has been completely emptied by the collector, the readout can continue.

The logics of a pipe is quite easy to handle as at any one time there is only one producer and one consumer for the data in each (half)pipe. If no collector or data sender is active, the readout will stall when the pipe becomes full.

7.1.2 TCP connections

With larger systems, where it is no longer feasible or desirable to have address-mapped databuses, the work of the `m_collector` program is divided into pipe reading on each read-out node by `m_ds` and event formatting by `m_dr`, and the distance in-between covered by commodity network. To handle large (instantaneous) data rates, several event builder machines running `m_dr` can be used in a N×M scheme, with peer-to-peer connections between all n read-out nodes and m event builders. Figure 7.4 depicts a multi-branch MBS system.

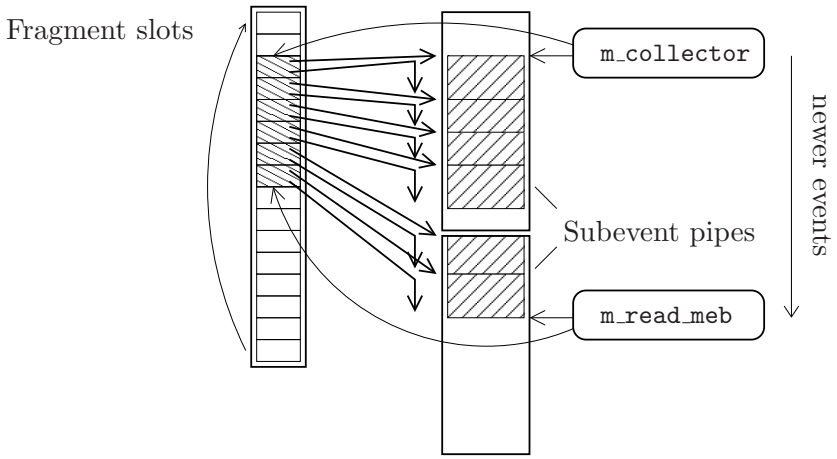

Figure 7.2: A subevent pipe has two parts: a configurable but fixed number of fragment (subevent) slots with control information referring to locations in a data buffer divided into two halves. The two halves are operated in a ping-pong fashion between the producer and consumer.

7.1.3 Buffers and streams

The next stage in the data transport chain is the buffers. An MBS node running `m_dr` or an event formatting `m_collector` has an array of streams. Each stream contains a number of buffers.

There is one producer of buffers — `m_collector` or `m_dr` — and several consumers — the stream server or event server and the transport process. The stream server sends data over the network while the main task of the transport is to write the data to disk/tape/RFIO, but it can at the same time also send the data over the network.

The streams are just a convenience aggregate of buffers used to keep track of which buffers are free, i.e. to be filled by the collector, and which contain data to be used by the stream/event and transport processes. The buffers are the same buffers that end up in LMD files, i.e. the stream and transport processes do no formatting at all, only copy the buffers. The buffer numbers are generated by the collector process. Figure 7.3 outlines a stream's life, cycling in the states:

1. Free/empty.

2. Filled by the collector.

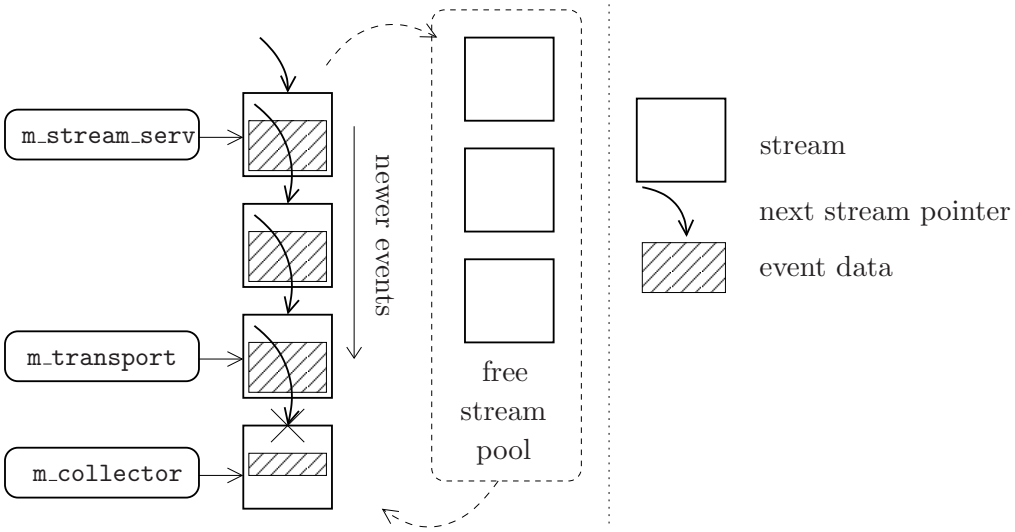

Figure 7.3: Streams hold fixed-size buffers of events. They are filled by either the `m_collector` or `m_dr` process. The data is then consumed by `m_transport` and/or `m_stream_serv`, during transport to storage or on-line use.

3. Waiting.
4. Used as copy source by the transport.
5. Used as copy source by the stream/event server.

Only one of the stream server and the transport is required. When one of them is not active, the other one knows about this and bypasses that stage to make the stream free again after use.

The event server is a replacement for the stream server that has the additional capability of only delivering events selected by a filter (specified by the `tcp` client) from the streams. It will only copy the wanted events from the buffers in the stream to the tcp output buffer.

As the buffers are of fixed length, some events will not completely fit at the end of each buffer. They are then split, with the beginning in one buffer and the end in the next (or several following) buffers. An event may however not span several streams.

The collector (and `m_dr`) has a keep-alive facility, to ensure that at least one stream is propagated every so many seconds, in order that even when events are produced at a small rate, they will still not linger around for too long, but be written to file. In this case, some buffers in the stream may

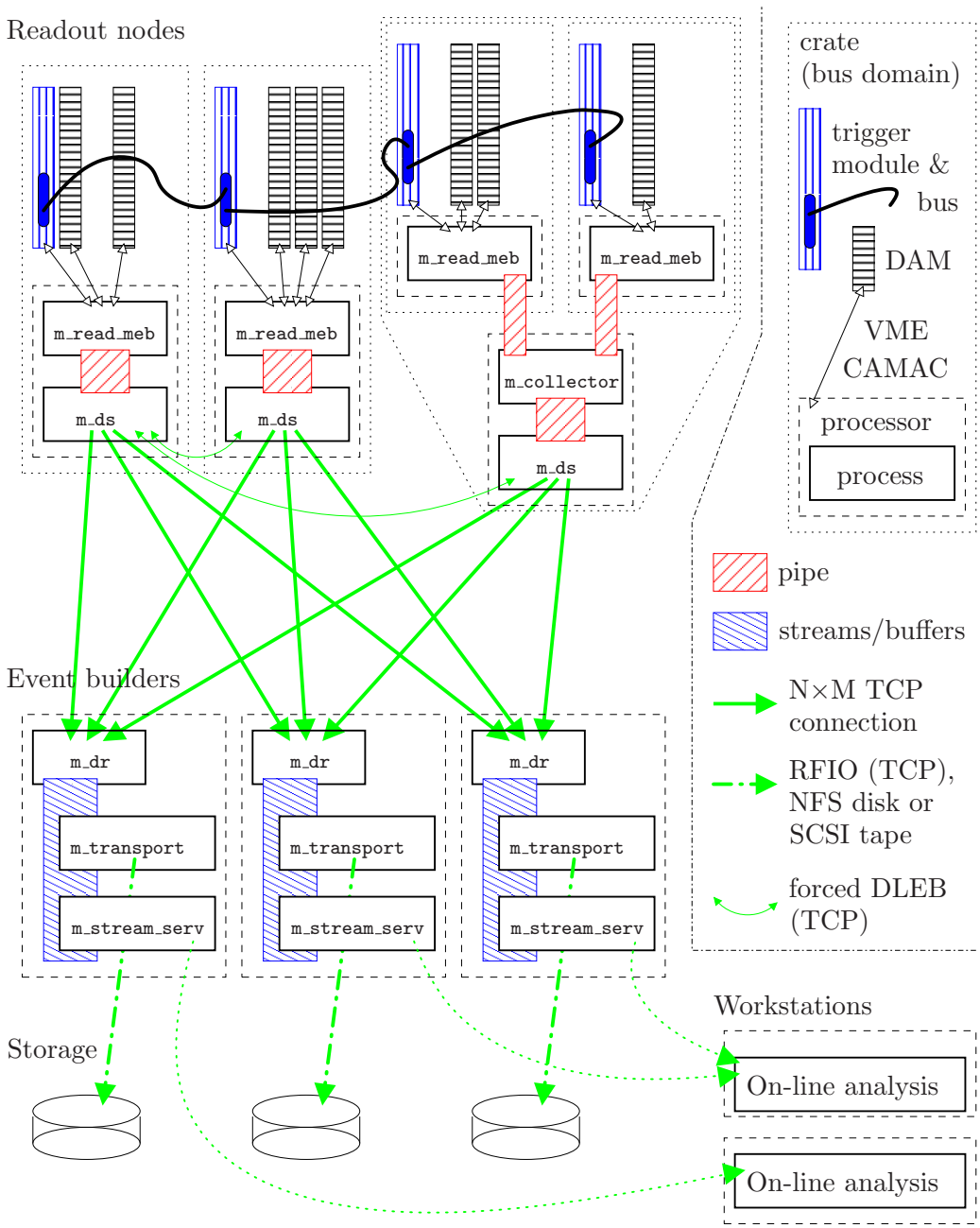

Figure 7.4: A full fledged N×M MBS DAQ system. The dead-time locking and triggers are distributed over the trigger bus. The trigger modules communicate with and thus synchronise the `m_read_meb` read-out programs. Data, written into shared memory, is propagated to permanent storage and on-line analysis by separate programs, combining data belonging to the same event, but progressing asynchronously.

be completely empty. The stream server will still send full streams, but the transport delivers only buffers with some payload.

7.2 Common event numbering with several event builders

Currently, when MBS operates in the N×M fashion, each event builder formats the events independently. The fragments (subevents) arriving from different data sender nodes are verified to belong to the same event, i.e. being synchronous (via the 5-bit local event counters and trigger numbers coming from each trigger module). All fragments for each event are packed into one event in the output stream. The event number is generated locally, such that with more than one event builder in action, there will be duplicate event numbers in the generated files.

Having one global event numbering can be quite helpful during analysis, and is sometimes necessary. To be able to uniquely assign which spill (accelerator cycle) an event belongs to, it is necessary to be able to track which is the previous (or next) spill on/off event. For events that have ended up in different event builders, determining such relationships is not possible without having an inter-event-builder ordering of the events. Another reason is when it becomes necessary to relate neighbouring events (most likely due to some DAQ malfunction which is detected during analysis), e.g. to compare scaler readings.

When calibration parameters are varying (slightly) over the course of the experiment, some markings of the events must be used to be able to use the correct parameters. A better alternative may be the use of a time, but also this time would then have to be the time of the event itself, and not the independent times produced by the different event builders. As the only time delivered by default by the MBS/LMD file format is the creation time of the buffers, not the events, this may be a minute off.

To provide a time is of course quite straight-forward. Just select one of the read-out processors (e.g. the one with the master trigger module) to create a time-stamp each event with the needed resolution (e.g. from the `gettimeofday` function) and place it in a subevent. The same could be done to achieve a global event numbering, i.e. let one read-out node produce an unique numbering in one subevent that appears in all events.

As opposed to the introduction of an always present time stamp into the LMD event headers, which would be a modification of the data format, making the existing event numbers unique is merely a slight redefinition.

Hence, there are some reasons to do this inside MBS itself.

The event numbers are already present in the headers. Thus, it makes sense to use them directly. Having the information in a subevent, requires the associated subsystem to always be a part of the DAQ, otherwise data collected without it (but with several EBs) will need special treatment.

Having the (unique) event number in the header, instead of somewhere inside the data, makes sorting and merging of events much simpler, as a merger program (e.g. `ucesb/empty`, see Write-up U) does not need to care at all about the contents of any subevents, but only inspect the event headers.

7.2.1 Implementation

Introducing the option[5] of globally unique event numbers is easy. Currently, each event builder has its own event counter, which is incremented each time the `m_dr` formats an event into a buffer. If instead the event counting would be handled by one of the `m_ds` processes (e.g. the first) and the numbers propagated to all `m_dr` processes, an unique ordering would be achieved. This works for the synchronous event building mode, where the data for each event from all `m_ds` processes is collected to the same event builder. In the asynchronous mode, where each `m_ds` sends data to any available `m_ar` (replacing the `m_dr`), there is no need to discuss event numberings, as event data is divided on purpose. The `m_ds` – `m_dr` protocol overhead is negligible, as only the event counter for the first event in each chunk of events must be sent — they all are in order.

A beneficial side effect of implementing this is made possible by having the `m_ds` keep track of the number of events it delivers, noted in the DAQ control structure residing in shared memory. Then, the `m_rate` program would be able to show the total number of events produced at each node, and with the addition of accounting of the amount of data sent, also the data rate from each node would become available.

7.3 Circular event pipe

The pipes used to transfer the fragment (subevent) data from the `m_read_meb` to the `m_collector` / `m_ds` are currently using a ping-pong logic for the locking between the producer and consumer processes. The data space of the pipe, acting as a FIFO, is divided into two halves, and only a (half)pipe that has been completely emptied may be filled again. This leads to effectively

[5] As some other experiments' analysis code may be depending on, or prefer individual event numberings, one should still support the old behaviour.

having only half the memory size set aside for pipe use available for buffering events. This limitation becomes most pronounced when using delayed event building, i.e. when the pipe consumer only propagates data along during off-spill pauses. When the spill goes on again, the event building will usually be suspended with one of the pipes almost, but not completely, emptied — making it unavailable for filling until the remaining data has been consumed.

By instead treating the pipe as a circular buffer, the entire memory can be used, as the locking between the producer and consumer instead carefully ensures that the producer does not re-use data space not yet consumed. The necessary accounting is two variables counting the number of bytes produced so far, and the number of bytes consumed so far. These are always increasing, with no need to wrap due to the wrapping of the circular buffer. The interesting quantity — their difference (the so far unconsumed amount of data) — is always less than or equal to the size of the pipe. Limited range variables, e.g. 32 bits, can be used even when the total amount of data that has passed the pipe exceeds 2^{32}, as the modulo-2^{32} arithmetics preserves the difference. This accounting is actually already implemented in MBS for the benefit of the delayed event building, such that it can know when to enforce event building even in spill. By using the variables `l_inc_wr_dat` and `l_inc_rd_dat`, it can avoid the situation where the (half)pipes become almost full.

Currently, the data of a single event may not wrap (i.e. fragment over the two half-pipes). Therefore, whenever the `m_read_meb` before the actual event read-out detects that the worst-case size of it would not fit in the remaining free space of the current half-pipe, it will switch between half-pipes and, if necessary, wait for the consumer to clear the other half-pipe, before starting to use it. Similarly, in the circular case, there are two reasons to wait:

- An event may not wrap the linear buffer space that makes up the circular buffer. Enough linear space must remain for each event to be read out. Otherwise, the remaining space will be discarded for use and the writing of the event will have to start at the beginning of the circular buffer's linear space. In case the entire end has not been consumed yet, `m_read_meb` will first have to wait for that to happen.

- When there is no longer any risk of overflowing the linear buffer area, `m_read_meb` will also ensure that the amount of free space is enough to hold an event, or it will wait for the consumer to proceed first.

One may note that both the write and read counters above need also to account for the discarded space as if it was used. This is easy for the producer, by just pretending to have written that amount of data. At this point, a

marker is also put in the next event slot, such that the consumer, when finding this marker, knows that before that event, the end of the circular buffer was actually discarded, and can do the same dummy accounting of the read counter.

7.3.1 Results

The modifications described above have been successfully implemented, and was in particular able to enlarge the available effective pipe memory of two slightly under-dimensioned systems (`r2f-6` and `r2-17`, doing FASTBUS read-out). These systems had received other speed improvements, which reduced the global DAQ dead-time, but still had rather limited physical memory available. They therefore posed the threat to not be able to cope with the higher data rates possible, causing a DAQ bottle-neck in some scenarios of the 2007 campaign, due to enforcing event-building in spill as the (half)pipes would be full before the end of each spill.

As the change only involved the `m_read_meb` and `m_ds` (pipe producer and consumer) it could be implemented as a slight modification on these processors only, with the remainder of the system using the normal production release of MBS. This was possible as the changes even reduced the number of variables needed in the MBS internal shared memory data structures.

7.4 Sporadic but long processing tasks

For a distributed DAQ with a global dead-time and many participating nodes, it is important to minimise the dead-time caused by individual systems, especially at times when the rest of the nodes would be ready to take another event.

In N×M configurations, the readout processors usually run only two programs that consume any noticeable amount of CPU time: `m_read_meb` and `m_ds`. Even though `m_ds` uses little CPU time, its actions (of sending data over the network to the event builders) can still adversely affect the average response time of the readout process, leading to unwanted dead-time. This comes about as the different nodes do the processing at different times, i.e. holding different events up. To avoid this in spill, MBS can be instructed to defer `m_ds` processing and sending data over the network until off-spill periods, using the 'delayed event building' mode, as long as pipe space is available (cf. Section 7.3).

Although this minimises dead-time during spill, the off-spill behaviour becomes worse, as more data must be handled there instead. But even off-

spill is not a good time to incur unnecessary dead-time, as this is the period during which calibration events often should be collected — muons from cosmic rays (utilised by e.g. the LAND detector), time calibrator events, pedestal events, pulser signals, etc...

The impact, either in or off spill, of the `m_ds` work is also almost proportional to the number of readout nodes. This is due to the network transmissions that work in a round-robin fashion, making different systems busy at different times (events). This makes it a scaling problem when constructing ever-larger setups, as already seen, and even more critical as R^3B approaches.

7.4.1 Wild goose chase: de-prioritising processes

With only two processes contending for CPU time, the obvious solution would be to use the real-time scheduling properties[6] of the operating system, and make `m_ds` run at a lower priority than `m_read_meb`. The read-out will then respond immediately when needed. As the read-out process does not use much time in total, `m_ds` gets a chance to run soon enough.

Unfortunately, for this to work some changes are needed to modify the blocking that ensures that the producer `m_read_meb` waits for the consumer `m_ds` to clear some pipe space when none is available . On these occasions, the high priority process must be made to wait for the low priority thread. Normal MBS behaviour is to wait using the `yield` system call (when `m_read_meb` and `m_ds` run at the *same* priority, the former can thus relinquish its CPU time-slice, giving the latter an immediate opportunity to do some work). The blocking is easily changed to let `m_read_meb` wait for a token in a file-system `pipe` by a `read` system call. The token is placed in the `pipe` as the result of a catching a signal sent from the `m_ds` process. Overall, the blocking is only utilised on the rare occasions then the pipe becomes completely full, and therefore not timing-critical. But unless working, the entire DAQ would lock up when this happens.

Much worse was the realisation that the priority elevation did not do the trick at all — the dead-time was not reduced appreciably...

7.4.2 Preparing network packets

After more intense investigations, particularly timing measurements of the various components of `m_ds`, the suspicion arose that it was not the processing *per se* which caused the sporadic but long periods of dead-time. The long

[6] With real-time scheduling, a higher priority thread immediately executes when it has work to do.

periods of non-interrupted work of `m_ds` are happening in kernel mode when the data is being prepared to be sent over the network, i.e. when `m_ds` has issued a `write` system call. Although the transmission as such over the physical cable is being done by the network interface hardware asynchronously to the CPU, there is some preprocessing done to insert the data into the network stack buffers. This work, being done in kernel mode, apparently operates at a priority higher or equal to the `m_read_meb`, making it not preempt immediately after the arrival of an interrupt from the trigger module, at which time `m_read_meb` has (urgent) work to do.

An explanation for this is that the network stack is used by many processes. Most of them are by default at the same priority as `m_read_meb`, resulting in the network stack being held at a higher priority than `m_ds`.

Semi-solution

Without altering the priorities, the problem cannot be completely solved, but the incurred dead-time can be substantially reduced. Make each `write` call issued by `m_ds` send a smaller chunk of the buffer, interspersed with yield calls. Thus, `m_read_meb` gets more frequent chances to execute. As long as `m_read_meb` does not do any long CPU processing it will also not affect the output data transmission, as the real bottleneck is the bandwidth of the network connection itself. In conclusion, the trouble with `m_ds` (`write`) is not that it takes much CPU time in total, but that it consumes it in too large quanta.

This modification has been tested and works as intended. With a system with 7 read-out nodes, the dead-time during delayed event building off-spill was reduced from an average of 1100 μs to 570 μs. The dead-time in spill (without event building) was 460 μs in both cases.

One caveat should be observed: chopping the insertion of the data into the transmit buffers into many `write` calls incurs an overhead per call, about 5 μs[7]. As long as the total CPU utilisation to saturate the network output (together with `m_read_meb`) does not exceed 100% it does not matter at all. At 5 μs and 4 kB per call, saturating a 1 Gbps link, the additional call overhead is 150 ms/s or 15 %.

Original approach revisited — real solution

An alternative to decreasing the `m_ds` priority is to instead increase it for `m_read_meb` just before waiting for a new trigger, and lower it directly afterwards. This way, it would at the time of trigger arrival directly respond to

[7]Measured on a 300 MHz RIO3 processor, running LynxOS.

the trigger and start the read-out. There is a catch though — as soon as the priority is restored, the kernel task of preparing `tcp` packets would kick in again, making the dead-time just as bad as before.

The only viable solution with priority modifications is therefore to permanently increase the `m_read_meb` priority above that of `m_ds`, running the risk of deadlock in case of some inadvertent infinite loop in the user function. That could however be fixed by also employing a watchdog thread/timer at higher or equal priority, which e.g. each second would check that `m_read_meb` has not gotten stuck, in which case it would lower the `m_read_meb` priority such that normal means of dealing with the hung system can be employed. The watchdog would identify a hung situation by observing the non-progress of yet another thread, running at a priority lower than `m_read_meb`.

Still, if running the `m_read_meb` and `m_ds` at different priorities, one must modify the locking behaviour of MBS to not use the `yield` system call from the higher priority thread to allow the lower priority thread to do some work (as it will not work), but instead wait for a signal within e.g. a `read` call. For the same reason, `yield`ing would no longer be effective inside `f_user_readout` to allow other tasks (like `m_ds`) to continue working while the user function waits for some data transfer to complete. Also such delays have to be ended by interrupts such that `m_read_meb` can wait without a busy-loop, if any other task should get a chance to do some work.

This modification was implemented and successfully used[8] in an experiment in 2008. The read-out process becomes very agile[9] and delayed event-building is essentially obsoleted, as in-spill network activity and processing overhead only marginally affects the dead-time.

7.4.3 Delays when the DAQ analyses data

An unrelated problem, but with similar effects, is that of occasional but lengthy data processing done by readout processes. One example is the rare but time-consuming processing of pedestals to keep zero suppression efficient (see [1], Section 3.3). Most such calculations happen seldom, and are usually done off-spill when possible. When performed at any earliest convenience, it is however likely that different subsystems will do such processing on different events, keeping the remainder of the setup idle. By making all systems that need to do recurring calculations perform them at the same time, the overall

[8] One bug with a stuck read-out had to be corrected during the run. Thanks to the debugger-friendly design of MBS, it was possible to pin-point the reason for the stuck locking-mechanism on the first occasion, and apply corrections.

[9] One may say that the DAQ finally has gotten "its priorities right", namely to primarily do read-out.

impact is minimised. The off-spill events are convenient points. Furthermore, each system should only consider each 2^nth off-spill event. Even if different subsystems have varying periods between the calculations, the occasions will be maximally aligned.

7.5 Network queues

Another scalability problem, purely network related, stems from the event builder(s). The complete events produced by the `m_dr` process are sent to storage and possibly also online analysis by the `m_transport` and `m_stream_serv` processes over the same network interface as the small but serialising data requests from the `m_dr` process to the `m_ds` processes, the latter running on the read-out nodes. The outgoing bulk data transfers are performed as soon as possible, and their presence in the transmit queue will therefore delay the more urgent requests for further data by the `m_dr`, as seen in Figure 7.5.

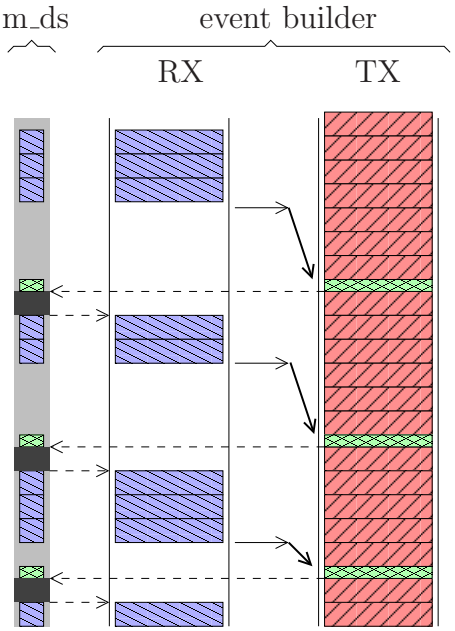

Figure 7.5: Delays caused by large buffers in the transmit queue of the event builder hinder transmissions to the read-out node.

The data request messages serve two purposes. On one hand to inform the various `m_ds` processes that a certain event builder is ready to receive data, and on the other hand to ensure that only one read-out node is sending data to each event builder at the same time. The enforcement of a single data transmitter at a time is necessary to avoid congestion at the network switch, in which case packets would be dropped and TCP would start the lengthy process of retransmissions, with large delays due to timeouts. As long as the transmission bottleneck is already felt at the output of the read-out node, no TCP back-off will occur. The only case when such a serialising scheme would not be necessary is if the network interface of the event builder would have more bandwidth than the combined bandwidth of the read-out nodes, e.g. a 1 Gbps interface receiving from less than ten 100 Mbps nodes.

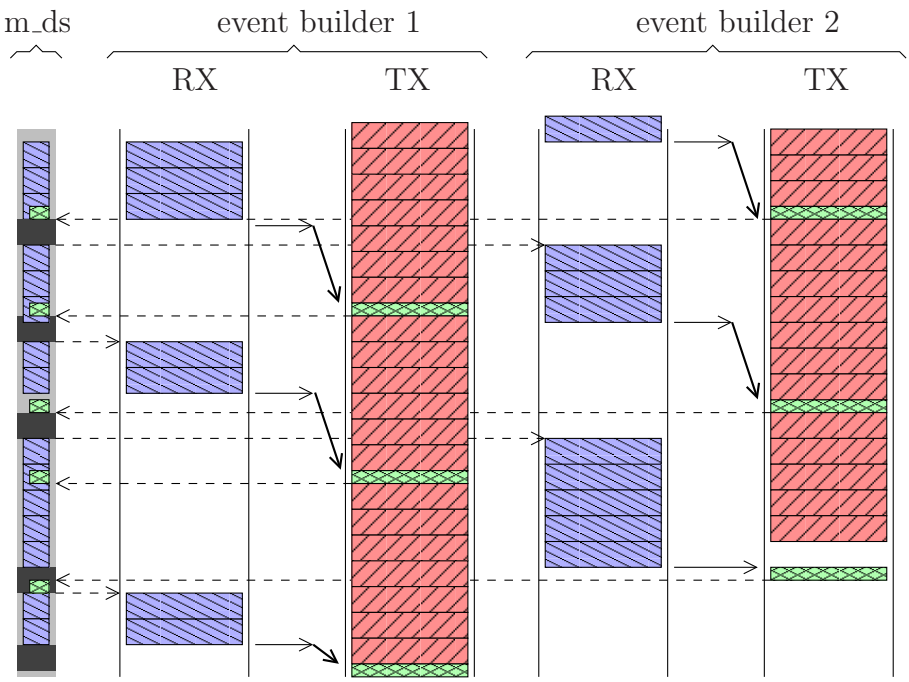

Figure 7.6: With enough event-builders, the transmission bandwidth of read-out nodes can be completely used even without MBS modifications.

The consequence of the serialising effect of the request packages is that between each received set of event fragments from each read-out node, the receiving end of the event builder will be idle for at least one network round-trip time, usually a few tenths of a ms on a modern network. With 5-10 read-out nodes and about 20 cycles over all nodes per second[10] this amounts to a few percent of the available time for transmission.

One should note that although this problem scales with the number of read-out nodes, as well as the more serious version described below, it only affects the efficient utilisation of the event builder resources. The transmission resources of the read-out nodes can be saturated by just employing more event builder nodes, as shown in Figure 7.6.

When the event-builder's transmit queue is filled with outgoing data by e.g. the `m_transport`, the request packets would generally end up at the end of the queue at the moment of creation, which with normal parameters may be a 3 ms long queue. With again 5-10 read-out nodes and some 20 cycles per

[10]The `N_FRAG` setup parameter should be as large as possible to reduce the number of cycles.

second, this amounts to several hundred ms per second of lost event builder input bandwidth. On average this would not be harmful, but is undesirable when employing delayed event building, where one would like to saturate the network during the off-spill pauses.

7.5.1 Hold transfer

The problem described above is an example of low latency communication being hindered by a high throughput transmission. Such a problem would generally be solved using a quality-of-service mechanism, e.g. the `ip(7)` socket option `IP_TOS` with `IPTOS_LOWDELAY`, to let the small but urgent packets bypass the long queue. This option however had no effect on the `LynxOS v4` installation tested, and on a `Linux` system, it could (with otherwise default network parameters) not recover to achieve even close to the request-response times attained in an idle network. Between two particular machines, with 100 Mbps connections, the round-trip time was 0.105 ms. On an unloaded network 9200/s `tcp/ip` request-responses (RR) were measured (round-trip time of 0.11 ms). By saturating the network link in one direction with a bulk transfer, the RR time increased to 1.6 ms (600/s). Using the "type-of-service" field to prioritised the RR traffic, the time decreased to 0.6 ms (1300/s). While better, it is still far from the minimum time, which probably can be attributed to the fact that while the urgent traffic was inserted earlier into the operating system's transmit queue, the final queue in the network interface card (NIC) itself has a certain depth which can not be easily by-passed.

The simplest way to decrease the latency for the `m_dr` request packages is thus to prevent the outgoing bulk data transfers from happening at times when the incoming data transfers are more important, e.g. during saturated data transfers during off-spill DLEB. Making this work dynamically requires communication of the momentary urgency level from the read-out nodes to the `m_transport` process, which is the cause of the harmful interference:

- With each chunk of subevent fragments sent, the `m_ds` process also sends along how many subevent fragments are currently pending in the pipe to be sent after this chunk. This is a measure of the urgency for the read-out node to get rid of data. If not enough events are in the queue to start transmission of another chunk immediately after the current one, it would not matter if the next transmission has to wait slightly longer.

- With each chunk received, the `m_dr` process records the local time at which it came and the number of further chunks to expect immediately.

It also keeps track of how many chunks were not seen since the previously received, i.e. how many other event builders currently appear to be active, receiving data.

For each packet, whenever more chunks are ready for immediate delivery than the number of event builders estimated to currently be active, the `m_dr` will put a request in a shared memory segment that any bulk transfer process currently active should presently avoid transmitting further data. For the cases when the `m_dr` for one reason or another does not receive a chunk of data which has no immediate follower, and thus would not remove the request to hold the bulk transfers, the note also contains an expiration time after which the hold request should be disregarded. The expiration time is based on the previously observed time between two consecutive chunks, plus a margin.

- Whenever the `m_transport` process is ready to send the data of another stream over the network, it first consults the hold request in the shared memory segment[11]. If the request has not expired yet *and* there are enough free streams such that event building can proceed even though data is piling up in the EB memory, the `m_transport` will wait (`yield`). Back-pressure, completely blocking the data transport and acquisition, is avoided as the transmissions are resumed before the event builder runs out of buffer space.

  As long as the `m_transport` program is running, even without any data recipient active, there is no need for the `m_stream_serv` program to do any similar handling, as any stream must first be handled by the `m_transport`.

The above was implemented and achieved its goal. Almost the full receive bandwidth of the event builders' NICs could be used during intense periods of event building in synthetic tests, compared to about 70 % without the modification.

[11] The `m_dr` and `m_transport` always run on the same machine.

Chapter 8

Slow control events

The proposal put forward in this chapter has not been implemented yet. It naturally follows the investigations of the previous Chapter, as the acquaintance gained with MBS was a prerequisite to devising the plan.

Normally, a raw data file from an experiment contains information about individual physical events that took place in the setup. To analyse the data, also ancillary information is needed. The supplementary information would include the status of the DAQ — parameters important for unpacking. E.g. pedestals currently in use to perform zero-suppression and perhaps even compression. It would also include slow control status, e.g. HV settings, magnet currents, thresholds...

At analysis, also other calibration parameters (gain factors, time synchronisation, effective speeds in the detector materials) are needed to interpret the data. These are however not known at the time of data collection, and should thus not be included in the raw data files (even with approximate values). But information which is exactly known (and cannot change), would advantageously be included.

8.1 What is a 'slow control event'

A slow-control event is an (sub)event, generated by the DAQ or by the slow control containing information about the state of the setup[1], that goes into the normal event stream. It has the special characteristic of being sticky, in

[1]The importance of arranging all information as invariantly as possible cannot be overstated! Although e.g. each HV channel may be known by the slow-control (and DAQ) by both its hardware identifier (e.g. rack, crate, slot, channel) and the associated detector channel (e.g. LAND, plane, paddle, pm), any information should be stored addressed by the hardware. This information is accurate — it is used to access the hardware itself. A detector name mapping may later turn out to have been wrong during the experiment.

that it is always delivered to all consumers (stream/event/transport server listeners, as well as all lmd files), even if the consumer connects or opens a file days after the (sub)event was created.

8.1.1 Non-incremental

Treating the SCSs (slow control subevents) as non-incremental simplifies the mechanics in all levels of MBS and analysis routines that keep track of them. When one wants to change one 'parameter' within what is stored in a particular SCS, the entire SCS must be re-issued. The SCS can then at all levels be kept track of by the normal MBS subevent identifier (type/subtype/ctrl/crate/id, 64-bits in total). Whenever a new comes, the old is by definition obsolete.

8.1.2 'Out-of-band'

The SCS are inserted into the event stream from the usual `f_user_readout`, but with a function call instead of the normal writing into the output buffer. `m_read_meb` will after each event (if any SCS were inserted) properly put it into the subevent pipe, with the appropriate control flags, to mark the special event. The convention is that the SCS by default take effect *after* the event that was just processed, and before the next event. This is compatible with the fact the previous event has been acquired and converted before the `f_user_readout` is called. As the dead-time is under control of the `f_user_readout`, it can make any changes (that are associated with the SCS updated data) take effect before any further events(triggers) are allowed.

8.1.3 As seen by consumers

Normal events have type/subtype 10/1, events containing SCS could have e.g. type/subtype 10/2 [2]. To any consumer, this means that all subevents contained in the event are slow control subevents. They obsolete any old SCS with the same tag (type/subtype/ctrl/crate/id), and should (if needed) be kept for future reference. An empty SCS would kill that type of control subevent.

Although the SCS in themselves are not incremental, the slow control events (SCE) 10/2 are. Whenever a new SCS is generated somewhere and inserted into the streams of data, only that SCS need to be propagated, as only the identically tagged SCS is invalidated. When a new consumer

[2] If that is in use somewhere, we'd say something else.

is connected/opened somewhere, they of course must get a large 10/2 SCE with all SCS that are presently in effect.

8.2 Getting information from the slow control to the DAQ

Before continuing, a slight digression on how the DAQ-slow-control communication can be arranged. On e.g. the master read-out node, an additional tcp server program is running, which handles the communication with the outside world, e.g. the slow control(s). It acts as a bridge to the `f_user` using shared memory. When the slow control wants to insert a SCS, it sends it to the `m_daq_slow_ctrl_agent` (TCP server), which as needed will insert the data into the shared memory and put a flag 'data to be handled'.

The data to be handled may of course also be a real slow-control request, e.g. to ask the DAQ to change some bias, so the agent could run on all read-out nodes. The DAQ will sooner or later (usually at the next event) see the flag, and handle it appropriately. If there are no triggers at all, the master agent may of course soft-trigger the DAQ to make it handle an event. This could be after e.g. a 10 ms timeout, to achieve a responsive slow control system.

When the command from the agent has been processed, the DAQ writes the response in the shared memory (receive buffer) and flags the agent. The agent may now tell the slow control that the situation has been handled, and give any response back (perhaps a temperature or bias reading). The DAQ has of course also created and inserted a SCS as needed.

This way, one can even make (big) slow control changes, like shifting the ALADiN current, while maintaining precise information, as in Figure 8.1. For the analysis of each event, as much and exact information as possible is then available, enabling the use of the hardest cuts thinkable. There is also nothing preventing a change to overlap with any other independent changes, ensuring scalability.

8.3 MBS propagating data

The intention is to attach to the existing data propagation scheme of MBS (see Section 7.1 for an overview) and extend it as necessary. The modifications turn out to be fairly small and non-invasive.

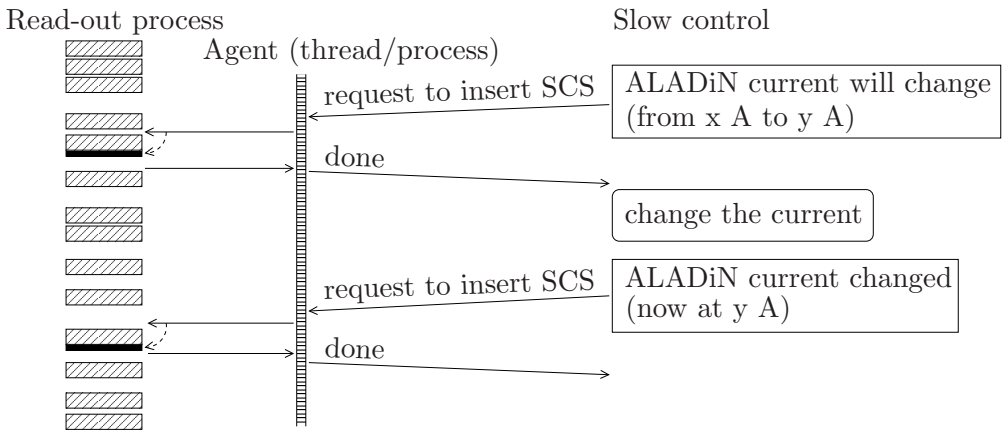

Figure 8.1: Inserting a slow control subevent via a slow control agent. Each step is simple, and efficient operation of the DAQ read-out itself is never jeopardised. The produced stream of events has an exact chronology of the changes to the experimental setup's setting.

8.3.1 Guaranteed delivery

To be useful, the delivery of SCSs must be ensured to all consumers. This is obvious for the case when they contain some parameters that have been used to compress the data, e.g. (rough) pedestal values for the SSTs, which are subtracted before compression, to flatten the data, thus improving the compression. Any analysis process must receive the SCS and the normal events in the exact order of generation.

This can be achieved by making every process that deliver data to anyone else responsible for keeping the most recent SCS of every kind. This is not as complicated as it sounds. Within the MBS, it will set a few rules for the handling of pipes and streams, which is where the data lives between the processing stages.

One important special case is that it may be of interest to be able to insert slow-control data also before the first event produced (after starting the acquisition), such that it can affect also that event. One may then as well make it possible to produce slow control subevents before and/or after any event.

An important feature is that it is the slow control *subevents* that are of importance, not the SCE. There should be no need to re-issue all SCS, just because one has been updated!

It is also important to note that the information in a slow-control event

never is associated with the event at which (before or after) it was produced. That event may be discarded due to some kind of filtering, but the SCS should not be. Unless one knows what one is doing, and can determine that the SCS is not of importance to any of the events accepted by the filter.

8.3.2 `m_read_meb` modifications

The `m_read_meb` operates with a fire-and-forget policy. As the pipe always is an one-to-one connection, when a SCS has been inserted into the pipe data stream, the `m_read_meb` may forget about it, the responsibility lies with the consumer, i.e. `m_collector` or `m_ds`. It is however necessary to prominently flag the presence of any SCS within the fragments.

As the `f_user_readout` function is called for each subevent with a pointer to write data directly into the subevent pipe, starting at the first free byte, the SCS cannot be written directly into the subevent pipe. There are two reasons:

- Without any special markings somewhere else, the SCS would 'get lost' within the fragments. Having the `m_read_meb` itself or the next stage `m_collector`/`m_ds` search for SCS would not be efficient.

- As the normal data is written directly into the pipe, `m_read_meb` has already made sure that there is space enough to hold the worst case fragment data (as declared in the `setup.usf` file). Direct insertion would require the worst case total SCS lengths to also be declared.

Instead, insertion of a SCS (which happens inside the `f_user_readout`, i.e. before `m_read_meb` knows the final length of the subevent), would be done by calling a function that will allocate memory for holding the SCS until the last subevent of the event has been produced by the `f_user_readout` calls:

```
create_slow_control_subevent(INTS4 **buffer, INTS4 length,
                             int flags, s_ves10_1 *header);
```

where the arguments

`buffer` is a return value to give a pointer where the payload should be written.

`length` gives the maximum size in bytes of the payload needed (such that the `f_user_readout` must not beforehand calculate the exact size, just an upper limit), `m_read_meb` ensures this amount as available after the `buffer` given back.

`flags` tells if the subevent should be inserted before or after the real event data. It could also have a marker if the length is exact, such that `complete_slow_control_subevent` below need not be called.

`header` (except the `dlen` member, the entry stating the length) is the subevent header of the SCS to be inserted.

After each SCS is filled, another function is called to tell the final size, by giving `buffer_end`, pointing to the next free byte after the payload:

```
complete_slow_control_subevent(INTS4 *buffer_end);
```

This will fill out the `dlen` entry of the SCS header. `m_read_meb` will ensure that the calls are properly paired and that the actual lengths used never exceed the requested sizes, i.e. that `*buffer <= buffer_end <= *buffer + length`.

After the last call of `f_user_readout` for an event, the SCSs are inserted into the data pipe by `m_read_meb`. Any SCS which should take effect after the event (common case) can just be added after the data already present, while SCS that are to precede the event (would be rare) requires the other normal event data to first be moved to make space. The memory allocated for temporary SCS formatting is kept by the `m_read_meb` between events, to generally avoid the overhead of memory allocation and deallocation.

8.3.3 Pipe extension

As the SCSs are using the same 'name space' as normal subevents (there is no bit in the type/subtype/ctrl/crate/id that can be unambiguously used to differentiate between a normal subevent and a SCS, they must reside in their own pipe slots. Those slots are marked to contain SCS data.

The SCS will also finally end up in separate events. Although the order is kept, the SCS are thus already within the pipes decoupled from the events with which they were generated.

8.3.4 Collector and `m_ds`/`m_dr`

As the slow control (sub)events are of interest to events *following* them, it would make sense for the collector process to handle them in conjunction with the next event, irrespective of their production.

As the collector (or `m_dr`) also operates in a fire-and-forget fashion, i.e. after formatting the subevents into a buffer/stream, it has no more to do with it, the responsibility of holding the data must be shifted towards the

streams themselves. (In order for the stream and transport servers not to have to take on the job also of keeping track of slow control events).

The `m_collector` (or `m_ds`/`m_dr` conglomerate) will each time it finds a SCS in any of the input pipes (or `m_ds`/`m_dr` tcp streams) before an event, format all the SCS from all sources into a SCE in the output and then format the real subevents into an event. An array (keyed on the 64-bit header) is also used to keep track of the SCS items, allowing easy replacement when new arrives. Before a new stream is started, and if any SCSs were changed during the previous stream, the current set of SCS items must be stored in a separate stream. In cases when the `m_collector` merely passes data along into a new pipe it need not keep track of the SCS locally.

8.3.5 Fast (critical) path and 'recovery' streams

In the steady-state operation of the DAQ, data is continuously flowing from the readout processes, via pipes, collector, buffer/streams to the stream server and transport process into a static set of clients and LMD files.

The exceptions (needing 'recovery') is when a new client connects (or a new file is opened), as they need a record of the complete set of SCSs in effect before the first real event passed on. Making the critical path fast, and recovery easy is simple with the stream/buffer design of MBS.

The SCSs will always be formatted in a SCE within the buffers in-between the normal events where they belong. That is all which is needed by any stream consumer (stream/event server/transport process) when delivering the steady-state data to their respective clients.

Whenever the consumer process sends a stream to a client which was not preceded by the previous stream (transport has opened new file, or stream server only delivering a fraction of the data), it will first have to send the 'recovery' stream/buffers, which contain the set of SCS in effect exactly before the new stream. The recovery stream has been separately prepared by the collector, see Figure 8.2.

When a stream/buffer has been completed by the collector, and if it contained any SCS, the collector will create a recovery stream (with one SCE) containing all SCS in effect at this point, before creating the next stream. Every stream created after this point, has a pointer to its preceding recovery stream. The chance that a recovery stream will be used at all is low, but it must remain until the last stream pointing to it has been declared empty.

It it up to the user to take into consideration that creating SCSs often will result in extra work for the collector proportional to the total size of the SCE, and a certain waste of stream memory used to hold recovery streams.

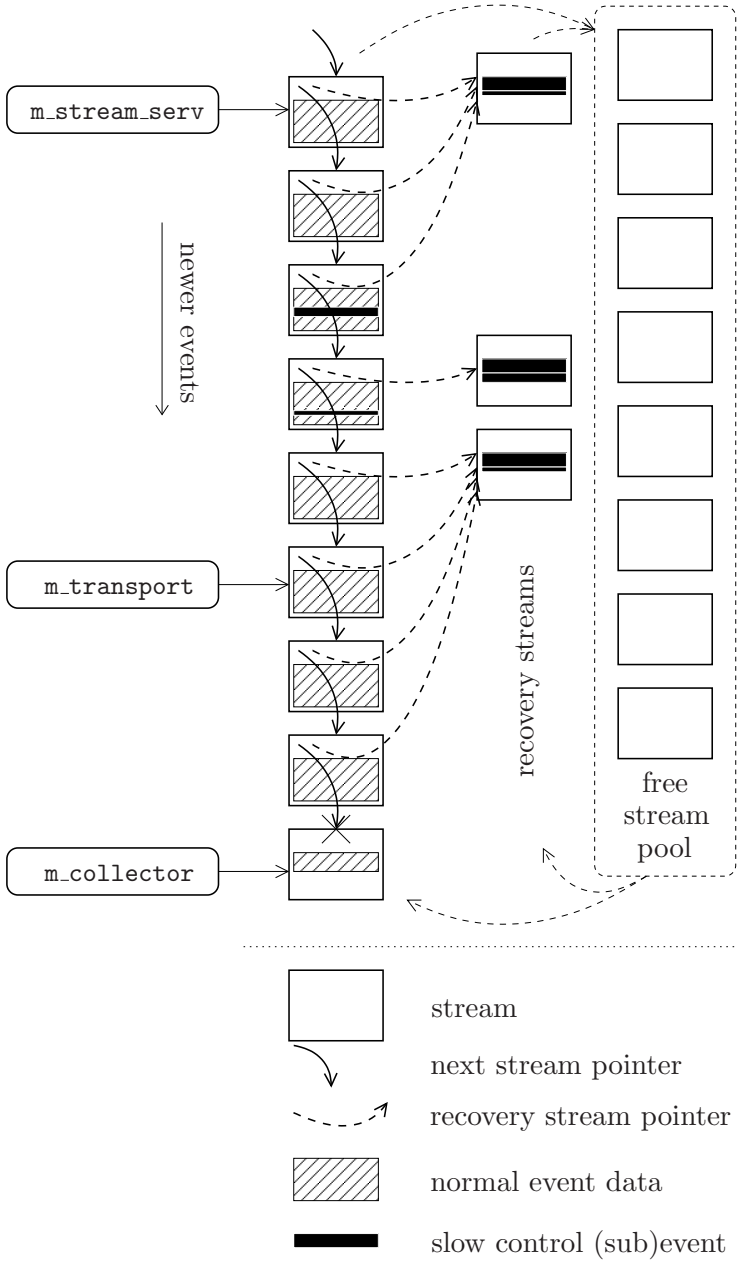

Figure 8.2: For each stream following a stream which contained a slow-control event, a new recovery stream (RS) is created. Each normal stream has a reference to the previous RS. The RS contain the full slow-control event status at the point before the new stream, and can be used for recovery before any following stream, as long as no new slow-control event appears. The RS is recycled when the last stream referencing it is released.

Performance and resource considerations

Reasonable use of the slow control event system would not cause any large resource consumption. One would expect the DAQ subsystems to at most create SCS events at the spill-off events due to updated conditions within the DAQ itself, which would be every few seconds. Slow control information originating from manual intervention (i.e. some human order to the slow control system) would also not occur more often than every few seconds, at most. Automatised varying of slow control parameters (as e.g. described in Section 8.2)) would also not create a flood of SCS items — even if a parameter has a tiny discretisation and all steps are recorded, data must be recorded during a time proportional to the number of steps to get enough statistics at each step, limiting the relative frequency of useful SCS insertions.

Pure monitoring data (like temperatures, actual voltages and currents) would generally also be rare, but should anyhow not be encapsulated in SCS but in normal subevents (even though they could be inserted by the DAQ via the `m_daq_slow_ctrl_agent` program from the slow control system). This since their values concern momentary readouts of various systems, and not parameters which are 'guaranteed' to be valid until the next update.

8.4 Open questions

How should the event numbers of the SCEs be handled? For the normal case, i.e. with SCEs included into the streams, they could follow the normal event numbering. Some event numbers would then simply refer to SCEs. As the SCSs are important — not the SCEs (and their numbers) — the primary requirements on the solution is to not break (or as little as possible) any existing analysis tools.

There are requirements from the N×M point of view, i.e. when several event builders are in use. They will then each deliver events in order to each stream. One should note that every event builder will see all SCSs, and that every LMD file will contain all SCSs. Therefore, normal SCEs may obtain ordinary numbers, but the number of SCEs created as recovery streams by each event builder is not known á priori and it is therefore without synchronisation[3] not possible to know many event numbers to reserve.

Still, any SCE inserted as part of a recovery stream, must also be assigned a number. Due to continuity of LMD files, it cannot use the same number as was possibly used in any previous file, i.e. the previous number. It would also be undesirable (and logistically complicated, particularly when implementing

[3]Which would largely defeat the entire purpose of the independent event builders.

unique event numbers over all event builders, see Section 7.2) to leave event numbers unused, just in case a recovery stream must be produced at some certain point.

It would be enough to allow the recovery SCEs to have the "magical" number 0 (or -1), as they within a file are logically attached to occur exactly before the next event. When the event numbers are used to merge the events again (in the presence of global MBS event counters), the SCSs coming from different files are handled with the normal replacement rules. (when a recovery SCE is found, one could in case no events actually are missing check that the SCS from the recovery event matches the accumulated SCS from previously processed data.

Buffer numbers

The buffers (in the streams) also have individual numbers. In the normal case, they are just continuously increasing. Empty buffers (due to stream flushing) obtain no buffer numbers.

How should recovery buffers be numbered? Fortunately, the buffers have no connections between different event builders, so only internal consistency must be ensured. Either the recovery buffers are numbered in the normal way, or they use 'magical' (unordered) numbers. The continuous numbering makes most checking possible. The drawback is that some buffers will appear to be lost, when in reality the lost numbers are just recovery buffers that did not need to be included. One may envisage a marker (or even count) in the following buffer, telling how many recovery buffers have been omitted before it. In any case, buffers that *do* contain SCSs should be marked, such that any program that skips buffers can know if it has lost any SCSs, and thus must use the recovery buffer. This gives an additional way to expect and verify missing buffer numbers.

Chapter 9

Absolute time calibration with a clock and NTP

Nowadays, it is common practice to synchronise the time on network-connected computers using the network time protocol (NTP) to an accuracy of a fraction of a millisecond. It can together with a clock and a scaler module be used to achieve time scale calibration better than well below the least significant bit on the full scale range of TDCs. The goal of time calibration is to determine precisely the conversion factor from the digitised binary values read out to ns, i.e. the bin width.

The method is similar to using two random sources (with known rates), but instead relies on a few oscillators (with between themselves arbitrary non-integer fraction frequency relationships) and counting their pulses in well defined time intervals. It is a complement to the use of a 'proper' (or more normal) time calibrator module, like the Phillips 7120[1], see Figure 9.1. It can also act as a replacement, but as it does have some inherent disadvantages (mainly needing more statistics for certain operations), it is better viewed as a supplement. In any case, it can use the same infrastructure to distribute the calibration signals, such that the marginal cost of implementing any of them given the other is small.

9.1 Background – why needed and useful

In the analysis of the S287 experiment, it was noticed that some time channels (particularly) of the TFW exhibits some non-linearities. To make a short

[1]The Phillips Scientific 7120 Precision Charge/Time Generator module [56] can deliver a start and stop pulse with a known differential timing, i.e. requesting a delay of a ns and then $a + b$ ns, the difference b ns is well-defined to within 2 % of the range selected.

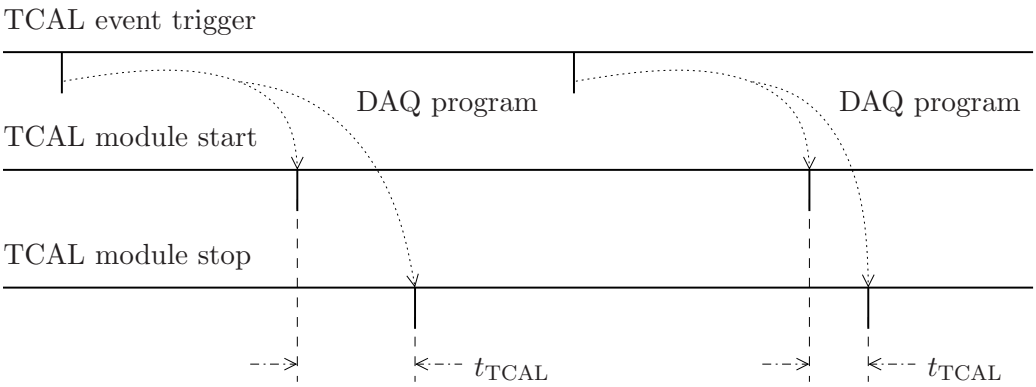

Figure 9.1: A time calibration module delivers a start and stop pulse with known intervals.

story long: it was observed that the position calculated from time behaved non-linearly. Similarly, the time-of-flight measurements were non-linear with respect to the reconstructed position. The effect could be diminished by applying a correction based on RAW level time values.

By looking at the time calibration curve, i.e. t_{TDC} vs t_{TCAL} (which should be, and usually is, quite linear), a slight deviation could be observed. This deviation however seems not to be in the part of the range that is used for the physics reconstruction, and could thus not explain the TFW misbehaviour observed. The author believes the actual cause to be a walk effect. The walk however is a devious beast, and difficult to pin down, as it continually defied our attempts at determining it[2] from experimental data [5].

The non-linearity in t_{TDC} vs t_{TCAL} is of interest anyhow. The time calibrator itself was checked using another channel (of the ZST detector) that fortunately had data in the same t_{TCAL} range. It was well-behaved, leading to the conclusion that the time calibrator module itself is not to the cause of the non-linearity in the time calibration curves of the TFW TDCs.

Even if it would not help S287, it would be interesting to have a method which with certainty can rule out (or find) any TDC non-linearities.

9.1.1 TDC non-linearities

Most high-resolution TDCs (least significant bit worth ≈ 25 ps) operate by charging a capacitor with a constant current between the occurrence of the start and stop pulses. The accumulated charge is then digitised with high

[2] I think we should call it Saruman [57].

precision as a voltage over a resistor using an ADC. If either the charging (or discharging) process or the ADC has non-linearities, this can show up as local compression or stretching of the time scale of the TDC. With 12 bits, i.e. 4096 channels and 25 ps bin width, the full scale range amounts to 100 ns — longer with lower resolution modes.

To counter non-linearities in the smallest bits of the ADC, most such converters employ a "sliding scale" technique, whereby a randomly chosen voltage is added to the voltage to be converted, and afterwards, the corresponding value is digitally subtracted again. This will smooth out any (small enough) non-linearities — at a slight cost of lost resolution, but only lost in case of non-linearities.

9.1.2 Counting/hybrid TDCs

Another kind of TDC operates with a frequency-precise clock for coarse timing, and some sort of interpolation between the pulses for fine-grained time information within the clock period. The clock usually operates in the region 40 – 300 MHz. Such designs are particularly suitable for multi-hit capable TDCs, as the "analog" interpolation only must cover a short time.

The common signal must in this case be handled like the other channels to be measured, or the clock distribution be common for all modules, lest time difference resolution to other modules become the clock period. The coarse timing is based on a simple difference of the number of periods counted to the arrival of the signal and common.

With a fixed frequency, no global non-linearities are possible. Within the interpolation network (in the form of short delay lines/stages, or similar to the technique above charging a capacitor) non-linearities are however quite common and may be considerable (several units in the least significant bit). They need to be corrected for to achieve the highest resolution.

9.2 Principle of method and implementation

By generating TCAL events with the start determined from one random source, and the stop determined from another random source, the times measured ($t_{\text{stop}} - t_{\text{start}}$) would be independent between events, and follow a simple exponential distribution. To not confuse the electronics, only one start pulse is allowed per event, and each time, the next stop pulse occurring after the start pulse is used[3]. The time distribution is determined by the rate

[3]Implementing this actually has a flavour of special relativity: what is simultaneous, or in this case, later. Later (or the next) is defined (at one point in space only) by a device

of the stop oscillator only, and given by how long time it on average takes for a stop pulse to occur after a start pulse:

$$P(t) = r \cdot \exp(-t \cdot r), \tag{9.1}$$

where r is the average stop pulse rate. The rate of the start pulses does not matter, we could even replace them with a periodic sequence of start pulses, and the result would be the same.

Let us turn things upside down, as random sources are hard to come by[4]. Let instead the start pulses be random, and the stop pulses given by a periodic sequence. In this case, the distribution will be uniform, over the period T of the stop signal:

$$P(t) = \begin{cases} 1/T & 0 < t < T, \\ 0 & \text{otherwise.} \end{cases} \tag{9.2}$$

If the period of the stop pulse clock is well known, the width of the bins can be inferred. For the overall gain factor (neglecting non-linearities), the counts in many bins can be used together, to reduce the number of events needed.

The idea is now to also replace the random start source with another crystal oscillator with a frequency unrelated to the stop clock. The start pulses will be random with respect to the stop pulses for events taken many cycles apart. In this case, the CPU clock of the processor controlling the DAQ will act as the start clock. For the particular case of the LAND setup, using TDC ranges needing a clock period T around 500 ns – 1000 ns, a 1 MHz clock serves well for the stop pulses, see Figure 9.2. Some special considerations will be discussed below to avoid some aliasing effects that could occur between the start and stop pulses.

9.3 CPU side: precise timing

The period of the stop clock can be determined by sampling a scaler counting the number of pulses at precise points over long time intervals (minutes). By sandwiching the command to latch[5] the scalers with the `gettimeofday`

at which both the train of start and stop pulses arrive.

[4]Radioactive ones are one, but they need detection, and for these purposes rather high rates. Other (pseudo-random) electronic sources face other instabilities (needing precise determination of the rates, and if they exhibit auto-correlations, the simple exponential behaviour will be lost).

[5]Most scaler modules have the ability to copy the current values from the actively counting registers to other registers simultaneously. The read-out of the copied values is not time-critical.

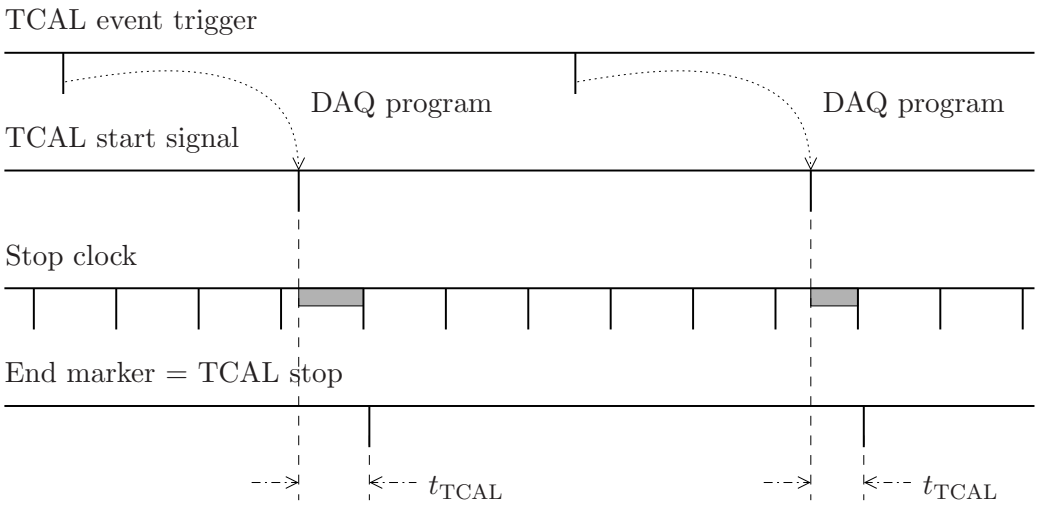

Figure 9.2: At the arrival of a TCAL start pulse, the next clock pulse is used as TCAL stop.

system call, the time of the latch is known with micro-second precision[6]. The time to do the read-out of the value itself does not matter, as it can be done after the second `gettimeofday` call.

```
gettimeofday(&before,NULL);
*scaler = LATCH;
gettimeofday(&after,NULL);

latch_at = 0.5*(before+after);
latch_uncertainty = (after-before); // even minus a constant
                                    // minimum overhead

... // do read-out of scaler values
```

It remains to ensure that the time given by `gettimeofday` advances as it should over the large time intervals. The oscillators used in computers are of moderate quality. A time drift on the order of a few hundred ppm (100 ppm corresponds to a 8.6 s drift/day) is not uncommon. By measuring the frequency error, it can be corrected for, and does not matter. Frequency fluctuations, mostly due to environmental reasons (temperature, air pres-

[6]On a RIO2/3, calling `gettimeofday` takes about 2 μs, and it returns values with full μs slicing of time. One may note that on a 300 MHz processor, each cycle is 3.3 ns. The time to issue a latch command over the VME or CAMAC bus is on the order of one μs (the former a bit faster).

sure) are worse. They appear on time-scales of hours and minutes, and are generally much smaller, a few ppm.

The drift and wander of the local clock can be determined with an accuracy of some fraction of a ms using the NTP protocol with one (or better: a few) external good NTP-stabilised computer clocks. To use the NTP protocol to get the current local time offset relative to a NTP server, one sends it an UDP packet marked with the current time (as a key, to later recognise the response). The server will respond by sending back a packet with the current time at the server (both at reception of the request and transmission of the response, so that the server processing time can be subtracted). By noting the reception time of the response, both the time difference and round-trip to the server (uncertainty) can be determined [RFC ntp protocol] (rx and tx are receive and transmit):

$$t_{\text{delay}} = (t_{\text{client,rx}} - t_{\text{client,tx}}) - (t_{\text{server,tx}} - t_{\text{server,rx}}) \quad (9.3)$$

$$t_{\text{offset}} = 0.5 * ((t_{\text{server,rx}} - t_{\text{client,tx}}) + (t_{\text{server,tx}} - t_{\text{client,rx}})) \quad (9.4)$$

By keeping track of the offset values over time, the local time as given by `gettimeofday` can be corrected (offline).

9.3.1 Gory details

The communication with the NTP server(s) is thus quite easy, but some care is required — primarily to not degrade the time information unnecessarily by introducing longer (apparent) delay/round-trip than necessary.

The key point is to i) get the local time just before transmitting each NTP UDP packet to the server, and to ii) get it again as quickly as possibly after the response packet has been received. The packets should also not iii) get stuck in the network stack transmit queue, on either client or server — where the client is the more likely one to, at times, have a long queue, due to the transmission of experiment data to event builders or other storage.

i) is easily solved, by doing the `gettimeofday` call just before the `send`ing the packet to the network socket.

ii) is almost as easy, by doing the `gettimeofday` just after the return from the `select` system call, which may indicate that a packet has arrived. The use of select is more or less necessary, and in any case convenient, as one must cope with both the situation that responses from the server(s) may be completely lost, and having a timer for making the queries (packet sending) happening at more or less regular intervals, as well as having contact with several NTP servers.

The CPU work associated with keeping in touch with some NTP servers is small, but requires the best response times available (when possible, i.e.

when the CPU is not busy with something else, i.e. normal DAQ processing). That is most easily ensured by performing this work in a separate thread of the DAQ program. Then, at those requests where (by chance) the CPU was idle at reception time, the round-trip delay will be at minimum. This also addresses problem iii). Those queries which suffered network congestion can simply be ignored in the later analysis by noting their larger delays.

In any case, the DAQ would only be responsible for storing the time delay and offset data (together with local time) such that a later analysis can do all filtering and trending (á la the NTPd servers) necessary to do the corrections for the frequency calculations of the stop clock.

9.3.2 NTP (daemon) particularities

When running on a computer, the NTP daemon requests at regular intervals time information from other computers (that are closer to a primary source, e.g. GPS or some other receiver getting data from some atomic clock) also running NTP daemons. That information is used to both adjust the local time (normally by slewing it, i.e. letting the adjustment happen over a longer time span, such that there are no jumps in time) and, if support is available within the kernel of the OS, to discipline the clock, so that it in the future should drift less.

The disciplining is generally performed in the kernel, and not on the underlying clock directly. By knowing the frequency shift of the clock, the values retrieved from it can be compensated, when any application need to query the time.

As the oscillator's frequency shift is moving with time (due to temperature etc), the correction values will also be moving.

The continuous corrections, and the fact that they are made 'after the fact', i.e. when the NTP daemon has received enough data to know that it is drifting leads to a slightly swinging behaviour of the local time. This is particularly the case when the daemon is restarted (e.g. after reboot) and if no or an old inaccurate estimate for the local frequency shift is available. From scratch, it seem to take several hours to reach a swing on the order of a few milliseconds, and about a day to get down to the tenths of a millisecond.

It is a good thing that the read-out processor is *not* running a NTP daemon, but only a script once an hour which queries a close by time server about how large correction is needed to get back to normal time (using slewing, i.e. calling `adjtime`). Thus, the local time will essentially have a saw-tooth shape when compared to an external, disciplined clock.

As we're not interested in the absolute time, but only the rate at which time passes, the fluctuations (derivative) become important and they will

determine the accuracy achievable. Assuming the read-out processor's clock to be stable during the time in-between its corrections, and that the time-scale of the fluctuations is several hours, with a magnitude of at most 1 ms, the relative precision would be 1ms/3600s, i.e. much better than 1 ppm, even 0.1 ppm.

To minimise the uncertainty introduced by the delay time when querying the NTP server, it makes sense to use a local NTP server that is close to the read-out processor (on the same network) even if that has a higher stratum than a more primary NTP server that is located further away[7]. More important is that the server used has had time to stabilise its own clock. The swing-in time is another reason to not run a NTP daemon directly on the read-out processor, as they may have to be rebooted occasionally[8].

9.4 Electronics side: random pulses

The extra hardware required in the LAND setup, which already has a time calibration infrastructure, is depicted in Figure 9.3. The stop pulse is taken as the end marker from one unit of a timer module. This stop is merged with the stop signal from the usual Phillips 7120 time calibrator. The signal starting the timer (enabling an end marker to happen at the next stop occasion) is created by the DAQ program, generating a pulse using an IOL-box[9], and is fed both to the timer as well as the master start (just as the 7120 time calibrator start).

For monitoring purposes, the TCAL stop is connected to a 'counting/hybrid' TDC channel, i.e. one with no integral non-linearity by design. In this case the LeCroy 2277 (which currently measures the pileup, being multi-hit capable) with 1 ns resolution suits well, as it has 64 μs range, much more than the envisaged 1 μs range of the generated random spectrum. The start signal to it is already delayed such that the spectrum will be within range.

The operation principle is: as the start pulse comes from the output of the IOL-box, its time will be determined by the last clock oscillator in the read-out computer (probably the memory subsystem) controlling the CAMAC interface. Since that oscillator has nothing to do with the oscillator generating the train of stop pulses, they will be random relative to each

[7]The round-trip delay is usually the same as the ping response time. Within the same network is usually about 0.2 ms. For a computer in the same country (the size of Germany), it reaches 5-40 ms, and to the opposite side of the globe half a second.

[8]Reboots are only rarely required, but when, then usually at the most inconvenient times.

[9]CAMAC module with in- and output connectors

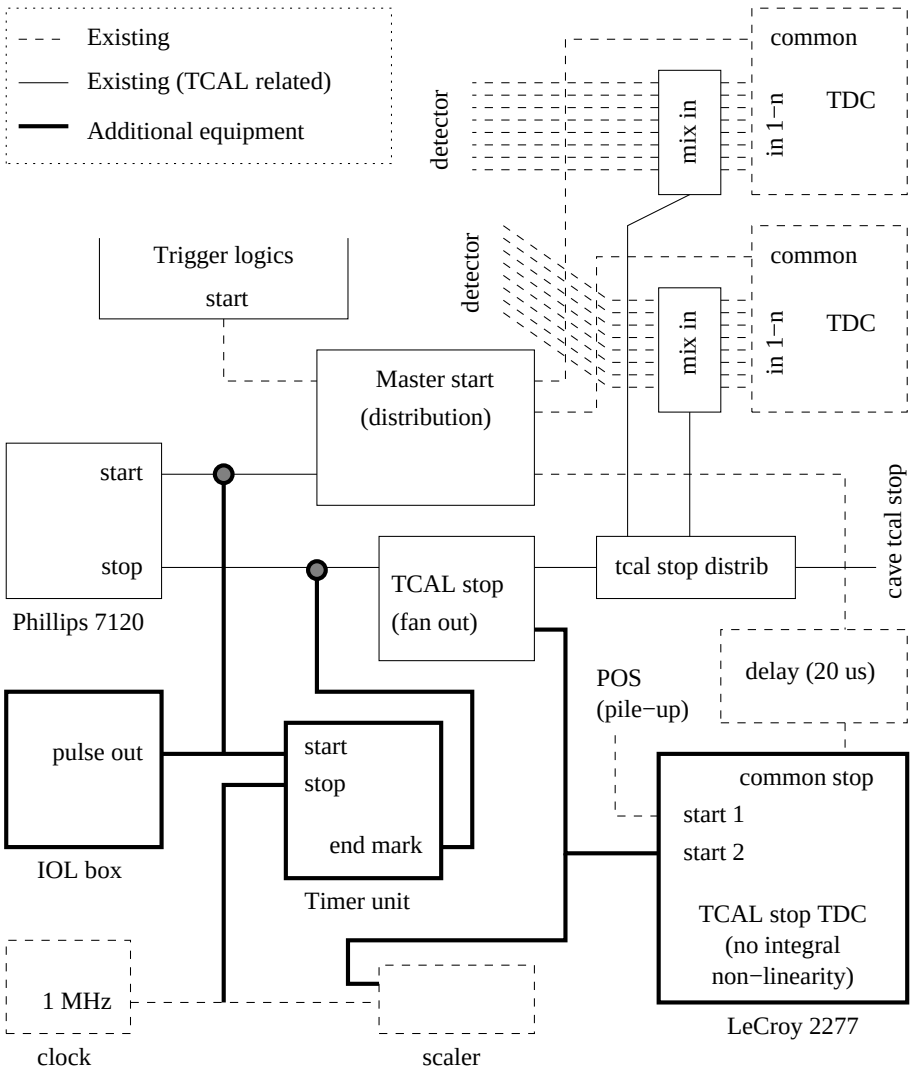

Figure 9.3: LAND electronics extensions. The times of the individual detector signals are measured with TDCs with common (start or stop) signals provided by the master start from the trigger logics. The existing time calibrator (Phillips 7120) provides a start and stop signal, the former fed into the master start distribution and the latter mixed (or'ed) into all individual detector channels. For the suggested additional time calibration scheme, the start generated by a pulse output of an I/O register and the stop is then caused by the next pulse of a regular clock, e.g. 1 MHz. Both signals are fed into the existing TCAL infrastructure. To measure the actual start-stop interval generated, the TCAL intervals are measured with a TDC having a range larger than the intervals covered by any TCAL method.

other. However, care has to be taken, as the trigger request for a TCAL event comes from a downscaled rate (e.g. 1 kHz) of the same clock as the stop pulses. The time for the trigger module, read-out processor and DAQ program to reach the point where the start pulse is generated will most often be around a quite reproducible but narrow time. That would cause a certain part of the 1 μs random range to be sampled more often than other parts, biasing the spectrum. This is mitigated in two ways:

- With the TDC that measures the locations of the TCAL stop pulses, any other spectrum (i.e. from any other TDC channel) can be normalised. This would be enough, except that statistics may become poor in some needed parts of the range.
- By letting the DAQ program randomly delay the issuing of the start pulse by a pseudo-random number of CPU cycles (each cycle is a few ns, total range on the order of a few μs).

9.5 Pros & cons

The proposed calibration method has some advantages and disadvantages relative to the use of a time calibrator module:

- As the method is based on randomly (relative to each other) occurring events and counting, it can *by design* not have any local non-linearities. Certain global effects may occur and must be handled, see the next Section.
- The counting couples the measurements to well-maintained external clocks (the GPS system or atomic clocks at e.g. the PTB in Braunschweig), giving an absolute time scale. One should note that the precision attainable is completely limited by statistics, though.
- A certain lower amount of statistics is required to achieve the precisions needed to compete with the Phillips 7120. This is due to the lack of the (for each event) á priori known programmed value of the time interval injected into the system.

9.6 What can go wrong?

The technique relies on counting the number of stops within many short spans of the TDCs' ranges. Some details need special attention.

9.6.1 Lost pulses

Assuming the randomness is good, the distribution function can still be defeated if pulses are lost. The TDC connected to the TCAL stop can monitor this. As it for each TCAL event generated is known by the DAQ program that it caused exactly one start pulse, also exactly one stop pulse should be observed. Thus, if e.g. the timer unit for some reason 'forgets' to generate the end marker, this will be noted. If no start pulse at all was generated, that would be noted as the TDC gets no common start, not generating any event.

If the clock generating the stop pulses for some reason occasionally looses pulses, that would be seen as the time to the next stop pulse then being within the range 0-2 μs instead of 0-1 μs, thus in about half of the cases, the stop arriving at the TDC would be outside the usual range.

Certain scalers (e.g. the LeCroy 4434, but not the LeCroy 2551 or the CAEN V830) may inhibit counting for a small time window around the latch or direct read-out of the counters, and would thus exhibit a dependency on how often they are sampled (possibly for other reasons, e.g. at normal events occurring). This effect can easily be measured and compensated for.

9.6.2 Distorted spectra

Theoretically, the distribution of the stop pulses should be a rectangle, such that within each time interval Δt, the probability to have a pulse is $f\Delta t$, f being the frequency of the stop pulse clock. This relationship would still hold in the interior of the spectrum, even if some effects may distort the behaviour at the edges (see Figure 9.4):

- Stop pulse jitter will cause some time spans to be unusually long and some to be shorter. This will smoothen the far edge of the distribution.
- If the timer is sensitive to the entire length of the stop pulse, and not only its rising edge, any stop pulse active while the start occurs will give rise to the same null time span, causing a peak at small times. The integral of those events at minimum times gives the timer's sensitive window, and they behave like taken from a part of the spectrum interior, reducing the total width. If this happens, the clock pulses should be reshaped to the minimum width accepted by the timer unit.
- Timer arming jitter and end marker jitter. In case it is not well defined at which time after the start pulse the timer is sensitive to the stop pulse, or simply the end marker is jittering, the edges of the distribution will be smoothed.

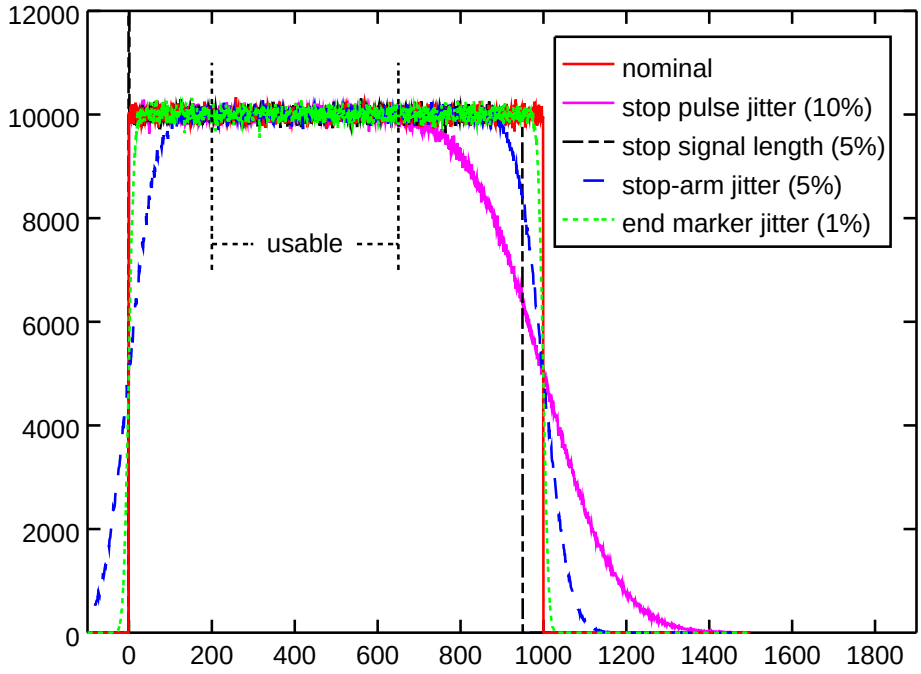

Figure 9.4: Simulated distortions effects of the TCAL stop TDC spectrum, albeit exaggerated to make them visible without zooming. Note that the location of $t = 0$ is arbitrary.

If the frequency of the stop pulse clock varies during the data collection, this would be handled as a superposition of distributions with those varying periods, in particular as varying probabilities to end up within a certain time span Δt.

9.7 TCAL stop TDC usage

The TDC recording the actual location of the TCAL event stop pulses make several things possible:

- Normalisation of counts in individual TDC spectra. When parts of the distribution is distorted due to the effects mentioned above, this is observed in the magnitude of the TCAL stop TDC spectrum, and thus

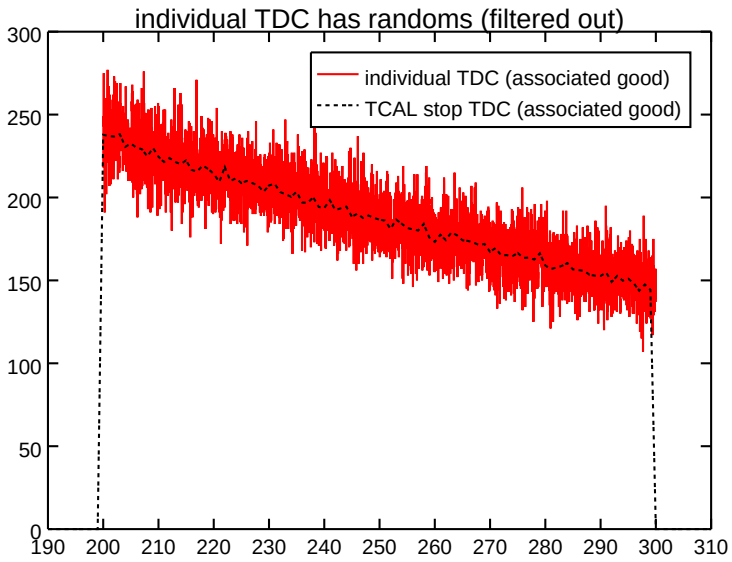

Figure 9.5: Simulated individual TDC time calibration spectrum loosing counts due to random background.

can be corrected for in the individual TDC spectra. This only requires a loose association between the location in the TCAL stop TDC versus the location in each individual TDC.

- As the individual TDCs are also connected to their detectors, they may observe spurious stop signals, that arrive in-between the start and the wanted TCAL stop pulse, thereby distorting the spectrum. Even though the chance of this happening is low (a random rate of 1 kHz, within a typical range of 200 ns has a probability 0.0001 to give the time of the 'wrong' signal, as only half the pulses will be earlier). It can be almost completely corrected for, using a narrow window around the loose association to the TCAL stop TDC as a selection, and is then handled by normalising the spectrum, that uncorrected if smaller towards larger times. Figure 9.5 shows a simulation of an uncorrected 25 ps/channel, 100 ns range TDC with a considerable random rate.

- It gives access to a (rough) TCAL time variable, i.e. such that after calibrating it, the times in individual TDCs can be plotted vs the actual time sent, just like when using the Phillips 7120 time calibrator module. This is helpful, as not only a single spectrum can be used, where its

width or cumulative sum / integral is the only usable variable, but also 2-dimensional plots are possible, with line fitting to determine the slopes, reducing the amount statistics needed.

Employing a zero integral non-linearity TDC with even higher resolution (like the CAEN V1290), having down to 25 ps resolution, would even make the system surpass the level of detail available with the Phillips 7120.

9.8 Statistics required

Using counting only (which is definitely required for the TCAL stop TDC), the accuracy achievable is determined by the precise knowledge of the stop pulse clock frequency and the counting statistics. With the frequency determined with errors smaller than 10^{-6}, it is of no concern. The counting error in each bin being $\sqrt{n}$, the relative errors will be $\sqrt{n}/n = 1/\sqrt{n}$.

With e.g. 1 million counts, the TDC gain factor can be determined to a 0.001 relative precision. At a 1 kHz TCAL event rate, this data is collected in 17 minutes. With a more modest rate during experiment of 25 Hz, it would take 12 hours. This is of course using the counts in many adjacent bins and summing them, as the gain is a global parameter. For a TDC with no integral non-linearity by design, it is no problem.

As the individual TDC usually have higher resolution and only cover a smaller range (e.g. 200 ns) of the random distribution, the statistics per bin will be correspondingly poorer. With a 50 ps bin width, and a million events, only 50 counts per bin would be accumulated. The usual application of a sliding-scale technique however smears any non-linearities, such that verifying their linearity on a bin-by-bin basis is meaningless. Combining e.g. 128 bins á 50 counts yields 6400 counts, having a relative error of 0.0125. With the 128 bins making up a 6.4 ns range, this corresponds to 0.08 ns. In other words, a 80 ps deviation from linearity could be observed at 1 standard deviation. More conclusively, at 5 σ a 400 ps deviation can be detected.

9.9 Current status

The random TCAL method for absolute time calibration was implemented in time for an experiment performed in the autumn 2008, and was operated alongside the time calibrator module, allowing the experimental analysis to proceed using any of the two methods.

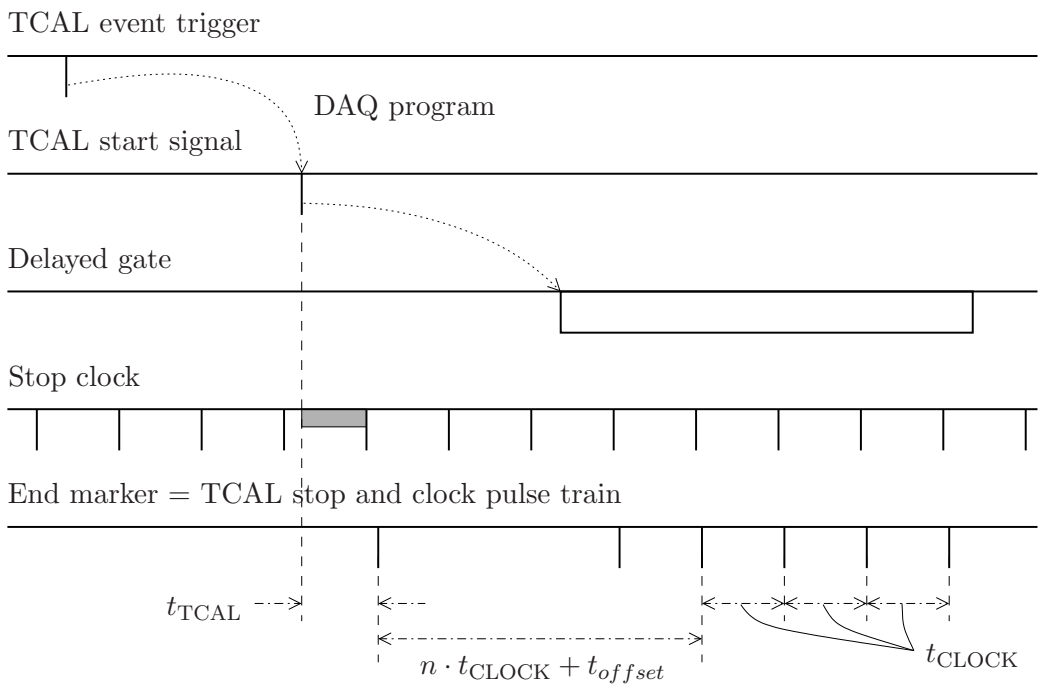

Figure 9.6: A gate is generated from each TCAL start pulse and and'ed with the clock to create a clock pulse train in the same channel as the TCAL stop TDC.

During the implementation of the additional electronics outlined in Figure 9.3 it was realised that one simple addition improves the method considerably while obliterating most of the needs for large amounts of statistics. As the TCAL stop TDC is multi-hit capable, samples of the clock period can be directly measured in the same events, by sending several clock pulses to the same channel that registers the TCAL stop. A sample of 4 or 5 pulses are selected by a delayed gate generated from the random TCAL start signal, as shown in Figure 9.6.

The first pulse of the sequence is not always reliable as it may be defined by the gate instead of the clock in cases where the gate start overlaps with a clock pulse. Since the clock period is known from comparison to NTP, the TCAL stop TDC can be immediately calibrated. Furthermore, the time between the TCAL stop and the clock pulse train should always be a multiple of the clock period plus a constant offset stemming from the two different propagation paths of the clock signal to reach the TCAL stop TDC channel. Events not fulfilling this criterion have suffered one of the effects described in Section 9.6.2, and are eligible for rejection. The TCAL stop TDC effectively

takes the role of a time calibrator module, providing t_{TCAL} as the reference value in each time calibration event. Other TDC channels can be plotted and fitted relative to this.

Preliminary analysis during the 2008 experiment showed the feasibility of the method. The absolute gain of our Phillips 7120 time calibrator module was determined at that time to be 0.25 % too small, well within the specification of ±2 %.

Chapter 10

Next step – tracker

This Chapter, partially in the form of an outlook, describes an ongoing work to develop a generic tracker for the experiments performed with the LAND setup. Although being primarily developed for use within `land02`, it turns out that most of the needed machinery is better moved to an external track preprocessor.

Tracking is the process of combining individual detector hits into the trajectories of particles (ions) passing through the setup. The interesting quantities are actually not the trajectory parameters as such, but the "quantum state" they represent — the initial/final values at the interaction point, i.e. momenta (or angle/direction and speed) and particle ID.

Independent of experiment scale or purpose, the principles of charged particle tracking are quite the same (with, of course, various ingenious exceptions). Rarely, one detector will by itself deliver all information needed to determine all the properties (mass, charge and momentum vector) of the ions entering or leaving the reaction target. After the hits in individual detectors have been reconstructed, their information must be combined to calculate the properties of the particles. Some examples:

- **Locations** are determined with position sensitive devices. They may internally determine the positions by:
  - being granular (strip detectors or wire-chambers with individual read-out),
  - charge-division between 'right-left' read-outs,
  - time-division by delay-line read-out,
  - drift of ionisation in gas detectors or
  - propagation delay and attenuation of light in scintillators.

The **direction** of a trajectory is determined by doing several position measurements.

- Time-of-flight measurements between detectors along the trajectory of particles give their **speed** (and energy if the mass is known). Alternatively, it can be found by employing a Čerenkov detector (which also can be sensitive to the direction). The curvature in a magnetic field is also linked to the velocity of a particle. The kinetic **energy** can also be determined by stopping the particle in a calorimeter, or by multiple sampling along its energy-loss curve while passing several material layers.

- The **charge** (Z) of a fully stripped ion is determined from its energy loss when passing through detector material, as given by the Bethe formula 1.1. Depending on what other measurements are available (if the velocity correction can be made) it can either be measured as an energy-loss ΔE when passing through a thin[1] detector or in a ΔE vs E fashion, whereby the total energy is also measured. See e.g. [58].

- The **mass** (m) of a particle is often measured as the ratio q/m by determining its deflection in a magnetic field (e.g. ALADiN). q is the charge of the ion; in the discussions of Section 10.2 ions are assumed to be fully stripped such that $q = Z$. The mass number A is roughly[2] equal to m (when given in u), with a maximum error of about 0.1 units in some regions of the nuclear chart. Alternatively, if the velocity is known, the mass can be determined by measuring the total kinetic energy.

- **Internal states**, like excitations, can be measured by gamma detectors (calorimeters) surrounding the target, unless the excited states are long-lived and escape the detection area before de-excitation. Spin polarizations can be measured statistically, by letting the ions scatter in an analyser (e.g. protons passing through a thick slab of carbon at the focal-plane polarimeter of the ESN at KVI).

[1] The thickness is relative to the ion energy and thereby range.

[2] This is a frequent source of approximate confusion when A is interchangeably used to also mean the mass, e.g. when giving a beam energy in terms of its kinetic energy per nucleon with the unit MeV/u.

10.1 From small to large, via huge

Different experiments have different capabilities and means of tracking, mainly depending on available energy, i.e. how much detector material the ions can be subjected to until the wanted information is washed out by reactions, straggling, or stopping:

10.1.1 Small: REX-ISOLDE experiments

Every μm of material in the target and detectors matters at reaction experiments in inverse kinematics at low energies, as in Paper I. For such low-energy experiments detection of the incoming ions is infeasible (a ^{4}He ion with 3.175 MeV/u has an implantation range of 100 μm in silicon).

Both the nuclide species and the momentum (direction and velocity) of light reaction products can be determined, by using a thin, segmented silicon ΔE detectors in front of a thicker silicon detector to measure the remaining particle energy. Measurements of the heavy (projectile-like) reaction products (or the incoming beam) are not practical due to the low duty-cycle — high rates during short spills ($\approx$ 100 μs every 10 or 20 ms).

The need to identify the incoming beam is mitigated by the relative cleanliness of the isotopes delivered by ISOLDE. Remaining contamination (mostly carbon and oxygen from REX-TRAP buffer gas) can, when not deselectable by m/Q in the post-accelerator, be subtracted as a background since it consists of stable isotopes and thus has no time-dependence with respect to the proton pulses impinging on the ISOLDE production target.

Although of course having its own complications [58], the track reconstruction for such experiments is straight-forward, not requiring (or allowing) much more than direct calculations.

10.1.2 Huge: Particle physics experiments (or FOPI / HADES)

With large and dense experimental setups like for example ATLAS at CERN, or FOPI and HADES at GSI, each particle is usually let to pass through many layers of similar sensor planes and volumes, leading to veritable *tracks* through the detector, see images at [59, 60].

While being large — often into the millions of signal channels — and having many tracks in each event — easily into the hundreds — the computational difficulties alone are impressive. But, at the same time, a reassuring regularity comes in handy. Whether a detector has 10 or 100 or 10000 channels, all behaving basically the same, should not matter too much. By the

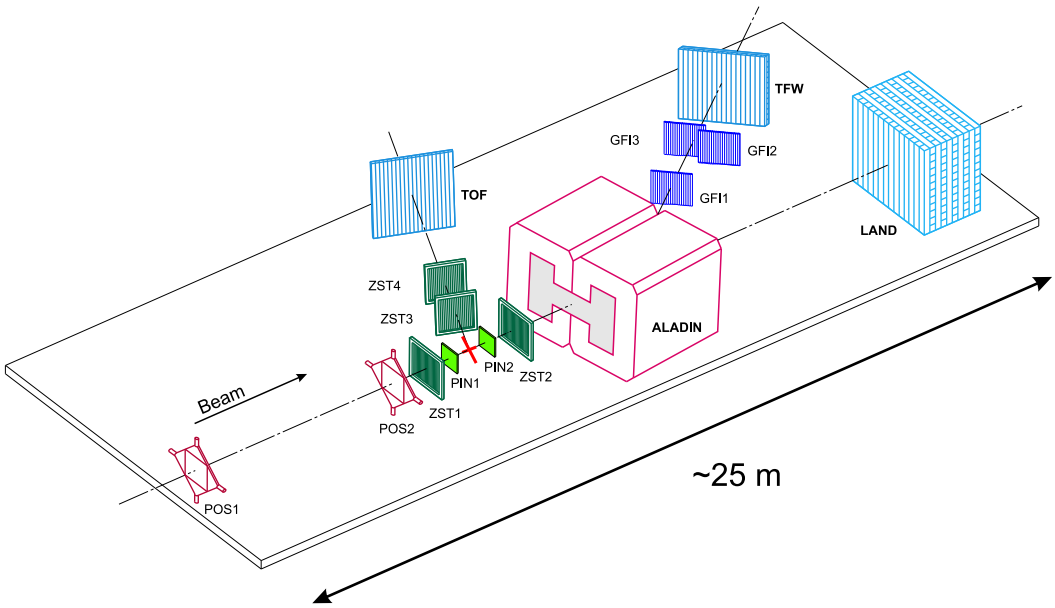

Figure 10.1: The ALADiN-LAND setup as used in S245. Several different detectors are used to determine the properties of each charged particle. Courtesy of [5].

use of computers, and their knack for performing repetitive tasks, one may expect the scientist's work involved to scale as $O(\log n)$, n being the channel count[3]. The handling of different kinds of detectors, which may be on the order of a dozen, is worse as each type usually requires its own special handling.

Trackers for these experiments usually take advantage of regular multiple hits in one or a few devices to form tracklets, which are combined to form complete tracks, also including information from other detectors that on their own cannot follow a particle.

10.1.3 Large: ALADiN-LAND / R^{3}B – in-between

The ALADiN-LAND setup borrows problems from both the regimes above. The setup is fairly large and complex, with many kinds of detectors, and the need to detect many different kinds of particles in a large dynamic range.

Up to now, the analysis of LAND experiments has not performed any tracking in its full meaning. A direct formula is used, which directly solves [61] for the wanted parameter (m/Z) of the charged fragment passing the

[3] If the effort scales as $O(n)$ or worse, one likely has a problem.

ALADiN magnet. The solution is calculated with respect to a reference track. The formula uses measurements of the x coordinate (magnet dispersive plane) of the particle in three detectors to determine the mass-to-charge ratio m/Z. Either one detector before and two after the magnet are used (the usual case) or two before and one after. In cases when more detectors have been available, the formula has been applied multiple times to produce an average. (Experiments performed by the ALADiN group, featuring a large MUSIC chamber behind the magnet, use a Runge-Kutta solver to account for the motion of ions through a measured magnet field map.)

The main reasons for introducing a general tracker are:

- Multiple tracks per event. The direct back-solved formula does not handle this well, not knowing which hits to combine.

- Gracefully handle tracks that for one reason or another are not detected in all detectors. (E.g. GFIs in S245 have low efficiency due to lightly charged ions.)

- Evict detector ghost hits. Ghost hits are produced when a detector with e.g. two independent perpendicular detection planes cannot determine which combinations are real in multi-hit scenarios.

- Handle all detectors and experiments with the same code. Differences in resolution etc. is to be handled by appropriate error estimates.

- Full fitting capability, taking all detector hits into account at once.

- Describe the particle motion accurately as they traverse magnetic fields (with detailed fringe fields) and matter causing energy loss. Needed for detectors with good position-resolution behind ALADiN.

- With most steps automatized, it may stand a chance to be easily used on-line for isotope identification and more advanced tests during the stressful situations that experiments present.

All this may sound a bit ambitious and over-kill. Many of the items can be neglected or handled as some overall correction in several experiments. The conviction and driving force behind this work is that instead of writing (and debugging!) many small simpler trackers with various optimisations, it is easier and more straight-forward to not negotiate with quality and to avoid approximations, especially if the extra work can be off-loaded to the computer.

10.2 Designing a tracker for a barely over-determined setup

The overall design of the data-flow in the tracker is given by the requirements mentioned at the end of the previous section. The following scheme evolves:

- The variables finally leaving the tracking stage are to be produced with a fit, optimised with respect to the detector hits. As a consequence:
  - The fit is a non-linear optimisation procedure, which needs start-approximations. These are delivered by a simplified (linearised) least-squares fit.
  - Ability to calculate residuals of the measured hit positions versus the positions implied by the parameters of the track at the target. Partial derivatives of the residuals with respect to the track parameters are also needed.
- Sorting out multiple tracks. Requires combining the correct sets of associated hits in the various detectors. This is mainly handled by the linearised fits for the start-approximation, by brute-force testing of virtually all combinations.
- Treatment of energy-loss and field-maps require explicit solving of the differential equations for the motion. With the fitting needing differentiation of the hit locations of each computed trajectory with respect to the start parameters, these combined requirements threaten to make the trajectory calculations calculationally prohibitive. This is addressed by separately preparing interpolation tables of hit positions as functions of the track parameters before event-wise processing is performed.

Each ion track is assumed to be completely described by its start coordinate in the target, momentum vector and particle ID using these variables (later referred to as ξ_i), see Figure 10.2:

- x_0, y_0 (and z_0) for the start coordinate. The z-axis is along the beamline at the target, x is horizontal and y is vertical. With thin targets, $z_0 = 0$. For thick targets, z_0 is handled separately, as the determination of the reaction point along the target requires information from multiple tracks.
- $x' = \frac{dx}{dz}$, $y' = \frac{dy}{dz}$, and $\beta = \frac{v}{c}$ for the momentum vector, i.e. described by two angles and the speed.

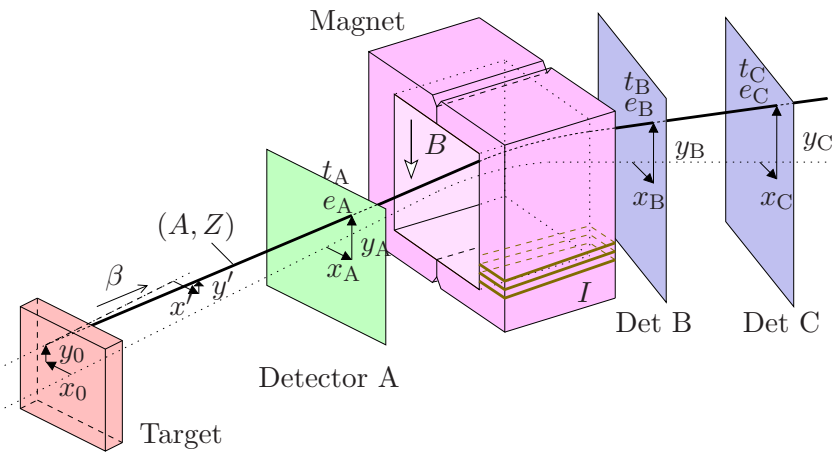

Figure 10.2: Initial parameters for a trajectory at the target (ξ_i: x_0, y_0, x', y', β as well as A and Z) and the variables describing intersections with detectors (V_k: x_A, y_A, t_A, e_A, x_B and so on). The magnetic field map B is selected by the magnet current I.

- A and Z for the mass number and charge of the particle, giving its ID as indices. (The real mass of each isotope, $m = m(A, Z)$ [62], is used in the trajectory calculations, and interpolated when A and Z are used as variables in the approximate preparatory fits.)
- I for the current of the magnet (e.g. ALADiN). (The current is one value, while the field is a position-dependent vector.)

During the final fit of a track, A and Z are constrained to integer values of existing isotopes. The magnet current I is only treated as a variable for optimisation during tracker calibration. It is a constant during individual runs of an experiment.

The hits in each detector are described and delivered by a few variables with estimates of their accuracy. The hit location (a and b with σ_a^2 and σ_b^2)[4], the hit time (t with σ_t^2) and the energy loss ($e = \Delta E$ with σ_e^2). For a RICH detector, also the velocity would be part of this list. Each detector in each event delivers values for the quantities that were measured. These are collectively named V_{measured}.

It is the job of the tracker to find and optimise the values of the first set of variables (ξ_i) with respect to minimising the discrepancies in the second set

[4]Each detector uses a local coordinate system with the origin in its centre. a and b correspond to x and y for a non-tilted detector perpendicular to the z axis.

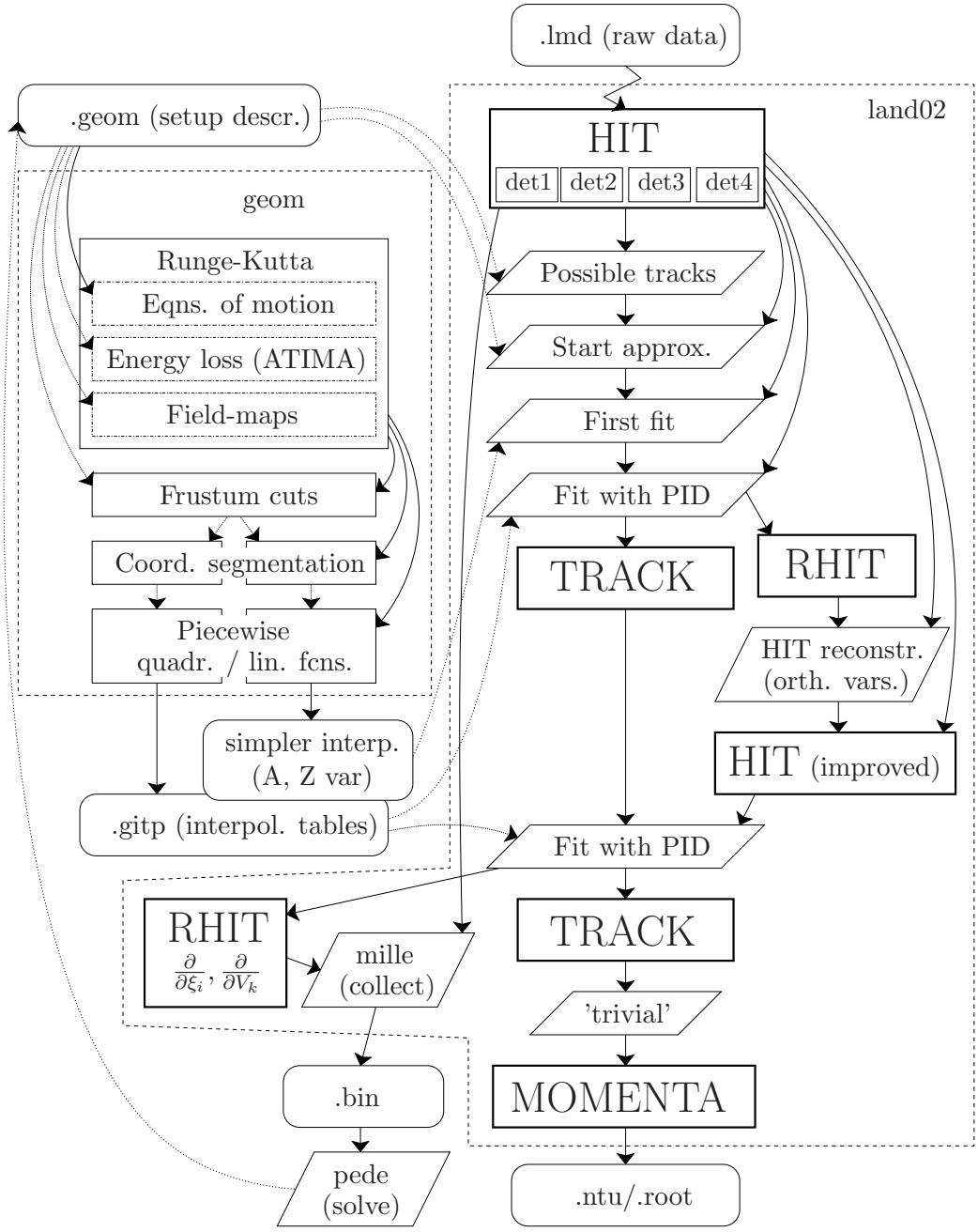

Figure 10.3: Tracking combines the hit information from many detectors to determine the ID and momenta of particles leaving (or entering) the target. The geometry of the setup is used directly to create start-approximations, and as input to a precalculation step (`geom`) which provides interpolation tables for the accurate track fits.

(V_k) for all involved detectors. Figure 10.3 shows the structure of the tracking work-flow, and is detailed in the following. The description is focused on the handling of outgoing particles. Tracking of incoming particles is much simpler, as e.g. multiple tracks are not considered[5].

10.2.1 Finding start approximations

Each track may only involve one or zero hits in each detector. A list of all possible hit combinations, i.e. all possible tracks, is generated on the fly by a depth-first loop traversing all hits, ordered by detector. The detectors are visited from behind, i.e. starting with the detector furthest from the target. Partial coordinate fits are updated for each added hit. As planes of limited aperture are reached, the fits are used to reject tracks that cannot have come from the target. If the hits collected for a track so far make the trajectory cross far outside the aperture, that particular combination is not built upon further.

When hits from all detectors have had a chance to be included, i.e. the target is reached, the least-squares fit for x, y and t as function of z is used to produce a start approximation. If a hit combination suggests a track which is not even closely originating at the target, it is ignored. The trajectories are in reality bent when passing the ALADiN magnet. This is modelled by an abrupt kink of the track in the least-squares fit, which makes it possible to derive formulas for direct evaluation, i.e. no costly optimisation stage has to be engaged. The kink approximation also involves a slight overlap or gap in the z coordinate at the magnet centre, see Figure 10.4.

These fits directly give the start approximation for the variables x_0, y_0, x', y', t_0 and β, and a value for the track angle after the magnet. The kink of the track is related to the curvature, and thus Z/m of the particle is estimated. An estimate of the charge Z is obtained from the detectors measuring energy loss of passing particles, adjusted with the approximate value of β obtained. Given Z, an estimate of m and therefore A can be determined.

The main point of this routine is to provide cheap start approximations for use by the next stages of more expensive optimisation, while finding all candidates for multiple tracks with effective, yet generous, culling of tracks (e.g. those partially caused by ghost hits). The main approximation made is that the speed of the particles does not change drastically while traversing the setup, i.e. that a constant β is sufficient.

[5] Arrival of multiple isotopes at the experimental target are in theory unlikely with the small rates we have. In practice they occur more often due to micro-structures in the long-spill extraction of ions from SIS, but must anyhow be removed due to pile-up in some detectors.

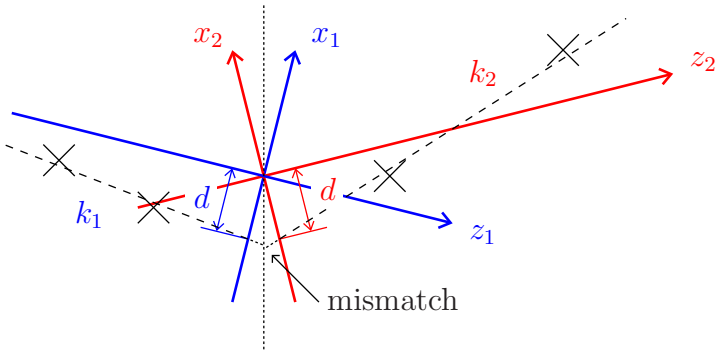

Figure 10.4: The start approximation for $x(z)$ (dashed line) is determined by a direct broken-line linear fit of the parameters d (intercept), k_1 and k_2 (slopes) to the hit locations (marked by crosses). Three hits are needed, with at least one on each side of the magnet. Attempting to repair the mismatch by joining $x(z)$ correctly at the magnet bending plane would result in unwanted non-linear equations for the start approximation.

10.2.2 Interpolation

The purpose of the track fitting is the comparison of actual and calculated hit positions, while the actual trajectory only is of interest as a means to determine the latter. With the initial parameters ξ_i and the calculated hit characteristica V_k in a particular detector, we are interested in the functions $V_k = V_k(\xi_i)$. They are just the integrated differential equations for the particle motion from the target to the detector, with forces representing the energy loss while traversing materials and the Lorentz force due to electric and magnetic fields. These functions are smoothly varying — otherwise fitting would not be possible. In order to avoid solving the differential equations for each trajectory in every event, the functions $V_k(\xi_i)$ are precalculated. One way to store and access them is via look-up tables, implemented as interpolation tables to preserve as much of the smoothness as possible.

The tables operate in a multi-dimensional space, spanned by x_0, y_0, x', y', t_0 and β, and at some stages also A, Z and I must be variable. The time-at-target, t_0, is just additive and can be immediately separated out and is not considered further. Using a piece-wise constant approximation over the necessary parameter-space offers no advantages over a piece-wise linear function in terms of storage of a look-up table, as one may as well directly interpolate linearly between the grid points. Furthermore, it would also completely destroy differentiability of the functions. Quadratic and higher-

order interpolation can be introduced to better approximate the function between grid-points, reducing the need for a fine-grained grid. Storage needs are reduced in exchange for more expensive evaluation.

With the starting point (x_0, y_0) only being variable within the confines of the target and therefore having only slight non-linear effects on each hit parameter (V_k), no segmentation is required and one expression without interpolation can be used (the index k is left out):

$$V = p_0 + p_{\mathrm{x}_0} x + p_{\mathrm{y}_0} y + p_{\mathrm{xx}} x_0^2 + p_{\mathrm{yy}} y_0^2 + p_{\mathrm{xy}} x_0 y_0. \tag{10.1}$$

The $p = p(\xi_i)$ are functions of the remaining ξ variables, of diminishing order with increasing order in x_0 or y_0, e.g. :

$$p_0 = q_{000} + q_{001} x' + q_{010} y' + q_{100} \beta + \ldots + q_{002} x'^2 + q_{011} x' y' + \ldots, \tag{10.2}$$

$$p_{\mathrm{x}} = q_{\mathrm{x},000} + q_{\mathrm{x},001} x' + \ldots, \quad \text{and} \tag{10.3}$$

$$p_{\mathrm{xx}} = q_{\mathrm{xx},000}, \quad \text{etc.} \tag{10.4}$$

At most, p_0 is quadratic, p_{x} and p_{y} are linear and p_{xx} etc. are constant. The qs are the interpolation parameters, above for 3 remaining variables of ξ, with each index representing the order of the corresponding variable. When also A, Z and I are variables of the interpolating function, there are six indices. In cases where less precision is needed, e.g. in initial fits, higher-order terms in both (10.1) and (10.2 – 10.4) can be ignored, reducing the calculational effort.

During fitting, the measured values ($V_{k,\mathrm{measured}}$) are compared to the results of (10.1) via the minimisation of sums over k of:

$$S_k = \frac{(V_k - V_{k,\mathrm{measured}})^2}{\sigma_{V_k}^2}, \tag{10.5}$$

with σ_{V_k} being the estimated measurement errors. The error estimates are delivered by the individual detector reconstruction routines, and may vary from event to event depending on the quality of each hit. And now comes the point where the interpolation tables justify their existence: During optimisation, derivatives of S with respect to the ξ are needed, and using Newton's methods to avoid tiresome line-searches, also second order derivatives will be needed. By application of the chain rule:

$$\frac{\partial S_k}{\partial \xi_i} = \frac{2}{\sigma_{V_k}^2} \frac{\partial V_k}{\partial \xi_i} (V_k - V_{k,\mathrm{measured}}), \quad \text{and} \tag{10.6}$$

$$\frac{\partial^2 S_k}{\partial \xi_i \partial \xi_j} = \frac{2}{\sigma_{V_k}^2} \left(\frac{\partial V_k}{\partial \xi_i} \frac{\partial V_k}{\partial \xi_j} + \frac{\partial^2 V_k}{\partial \xi_i \partial \xi_j} (V_k - V_{k,\mathrm{measured}}) \right). \tag{10.7}$$

Table 10.1: The number of floating point operations needed to evaluate a few common expression trees for the interpolation of each S_k and its derivatives.

Description	Order	Mult+Add	Used
8-variable (full)	quadratic	7281	never
6-variable (no A, Z, with I)	quadratic	1233	alignment with unknown I
5-variable (no A, Z, I)	quadratic	451	normal tracking final fit
7-variable (no I)	linear / simplified	161	normal first fit

The derivatives of V_k are easily obtained from (10.1 – 10.4), and will look similar and use the same interpolation table parameters. They are favourably evaluated together with V_k and S_k.

The evaluation of the resulting expression trees is suitable for evaluation with the parallel floating point units of modern processors and graphics cards. As these thrive on doing many *exactly* similar calculations at the same time, extracting the most parallelism will require evaluations for several events to be scheduled together. Table 10.1 shows the number of floating point operations involved in a few expression trees.

10.2.3 Track preprocessor (`geom`)

The interpolation tables are created before event reconstruction by a separate program, named `geom` after the fact that it must take the geometry of the setup into account. The setup is described as a set of planes with matter in-between, see Figure 10.5. The planes are essentially perpendicular to the beam. The active volume of a detector begins at one plane and ends at a later one.

The differential equations of a trajectory are integrated by an adaptive 3$^{\text{rd}}$ order Runge-Kutta method. Energy-loss tables are retrieved by calling the external program ATIMA [63] for the isotope-material combinations encountered. The magnetic field of ALADiN is described by a measured field map [64], measured for ten different current values and is interpolated in-between.

By having the capability to calculate any V_k as a function of ξ_i, the track preprocessor can perform all steps necessary to automatically produce suitably segmented interpolation tables:

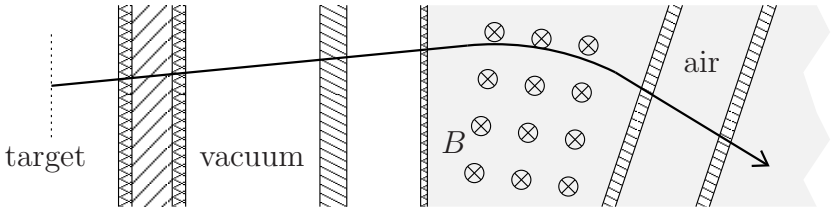

Figure 10.5: As far as the track preprocessor is concerned, the world consists of magnetic fields and a list of planes with materials causing energy loss in-between.

1. The ranges of x_0, y_0 and β, A, Z and I are given by the user. Determine the useful limits of the remaining ξ_i (x' and y') by finding the values for which the trajectories go outside specially defined frustum areas. A frustum is a rectangular or circular acceptance cut imposed by the setup, e.g. the beam-pipe or the ALADiN aperture, associated with a plane in the geometry description.

2. For each variable ξ_i (except x_0 and y_0) the required segmentation is determined such that the interpolation error in any V_k does not exceed one tenth of the expected detector resolution of that variable (σ_{V_k}).

   The order of the interpolating function can be manually limited for a particular V_k. As an example, the hit coordinates (a,b) at a detector directly behind the target without any magnetic field in-between will only depend on (at most) a linear combination of x_0 and y_0.

3. Ion trajectories are then calculated for the full ranges of all variables, forming a grid with several subdivisions in each multi-dimensional box of the interpolation lattice, and all V_k are collected. The interpolation parameters for each V_k are then determined such that the average interpolation error is minimised, through a global optimisation of each interpolation function. This leads to overdetermined sparse least-squares equation systems with the interpolation parameters as unknowns, solved by LSQR [65]. The equation systems are set up such that the interpolating functions become continuous.

This procedure can take about 15 minutes per isotope on a contemporary computer. Interpolation tables for different isotopes are independent and can of course be produced in parallel.

For a tracker of incoming particles, it is the track parameters as the ion *reaches* the target that are of interest and form the set ξ_i. In this case,

`geom` simply evaluates the trajectory of particles backwards from the target, "accelerating" them as they pass through matter.

As a side effect, the `geom` facility for calculating trajectories can be used directly without the full procedure for generation of interpolation tables. For simple lists of materials it can even be completely run from the command line, serving as a modern-day back-of-the-envelope calculator or slide rule of stopping power for a busy physicist.

10.2.4 Track fit

The purpose of the first track fit is to improve the start approximation values by removing the errors introduced by the linearisation, mainly the track kink and constant β. A and Z are still continuous variables, and the fit is performed with less accurate interpolation tables, with larger average errors, to reduce the calculations.

With more reliable parameters, further rejection of unrealistic tracks becomes possible. As the least-squares fits now include reasonable values for the sum-of-errors (S), these quality-measures can be used to compare tracks relative to each other for culling of less healthy specimens. A margin for the modifications that will be introduced by correction of the still unphysical A and Z must be allowed.

Accurate tracking

The still surviving tracks are then subjected to fits with real A and Z. Accurate interpolation tables are calculated for ions with physical (integer) mass numbers and charge. For each track from the previous stage, integer values surrounding the fitted A and Z values will be chosen and attempted. The distinct differences of tracks based of different isotope assumptions will lead to some of these candidates being much worse.

Within each event, a list of multiple tracks will be chosen as a set of tracks that uses the available hits in the detectors efficiently, in terms of least sum of individual S values. Several lists (using the hits in different ways) can be produced per event, each with an assigned probability relating to the quality of the included tracks. All hits must not necessarily be included in all lists, as they may represent noise.

Feed-back loops

Some detectors can not provide all V variables themselves, but would be able to do more accurate hit reconstruction if knowledge of the other variables

associated with a track hit were available to them. One example would be drift chambers, that with precise knowledge of the time when the particle passed the detector could do better drift-time estimates, or may use the angle of the ion traversing the detector to correct the drift-length. Another example is scintillating fibre detectors with single-sided read-out that could give a coordinate along the fibre, given the time of the ion passage, or conversely give a better time given the coordinate.

To enable such hit reconstruction improvements, a full set of the calculated hit characteristics at each detector for each track is made available to the detector reconstruction routines for improvement. This is done after the full accurate track fits have been made and good tracks have been chosen. The calculated hit parameters take the form of reference hits (named RHIT in Figure 10.3), and can be used by each detector to, when possible, improve on variables that are "orthogonal/independent". The improved values are then used to make one final fit of the chosen tracks. The independency requirement is to avoid formation of self-fulfilling "loops" that could make the track fit reinforce itself. The trivial case would be to use $V_{k,\mathrm{RHIT}}$ to calculate a new value of itself (V_k).

10.2.5 Calibration

The calibration parameters of the tracker are the locations of the detectors. Ideally, these would be known from measurements. However, reference points for the active areas of a detector are often hard to determine precisely, such that final alignment is best done with physics data. The old "tracking" method using direct backwards calculation simply defined a reference track through the setup and did all track measurements relative to this. Such a procedure is no longer possible, as it often involved effective rescaling of detector sizes and positions. But a similar and more comprehensive approach can be employed: `Millepede II` [54, 55].

As introduced in Section 6.2, `Millepede II` first collects (`mille`) data for alignment and then calculates (`pede`) a solution. For each track it requires:

- the differences between the measured and estimated hit variables, with error estimates, i.e. just $V_{k,\mathrm{measured}} - V_k$ and σ_{V_k}.
- the derivatives of the variables describing the hits, V_k, with respect to the local (event-dependent) parameters, ξ_i. These are just the $\partial V_k/\partial \xi_i$ discussed in (10.6) and (10.7).
- the derivatives of the variables describing the hits, V_k, with respect to the global (calibration) parameters g_j, i.e. $\partial V_k/\partial g_j$.

For detector movements in the plane of the detector, i.e. along the axes of a and b, the differentiation is trivial. Movements along the beam-line primarily affect the time-of-impact, and $\partial t/\partial z$ can be approximated by knowledge of the velocity of the ion when passing the detector. This, or described as its speed and angle of incidence, will therefore also be needed as interpolated values for calibration.

The energy loss generally varies only very slowly with the position of a detector, and would only in special cases be useful for detector alignment. It will often also be the case that the location of a detector along the beam-line is ill-determined with physics data, i.e. effectively causing singular alignment equations, and therefore must be taken as given by measurements.

The precise generating current of the magnetic field also often has to be determined from track data. This goes directly as $\partial V_k/\partial g_j$, with $g_j = I$ and thus calculated as $\partial V_k/\partial \xi_i$, requiring interpolation tables with variable I.

10.2.6 Current status

The frontier of this project is at making use of the interpolated tables to do the track fits.

With the completion of the tracker, `land02` will reach its goal to span the entire bottom arc of the lower circle in Figure 1.3, transforming data from a DAQ into variables for evaluation of invariant masses and cross sections.

Chapter 11

Future – outlook

The ALADiN-LAND setup has been operating since 1990. With recurring enhancement and upgrade efforts, the capabilities have been continuously extended. The most recent additions, two systems for detection of forward focused and recoil protons, were commissioned in 2007. The complexity of the setup and following analysis has expanded along multiple axes: number of channels, number of detectors and acquired data rate.

The ongoing developments which gradually will transform the setup into R^3B, fuels these trends. The continuous improvement of computing hardware, self-fulfilling Moore's law [66], can only partially help. Software does not lend itself to multiplication as easily as hardware[1]. Care, commitment and consolidation will be evermore needed.

11.1 ALADiN-LAND setup growth

The growing number of channels in the setup is shown in Figure 11.1. With LAND having 800 channels already in 1990, the growth has been relatively moderate until 2007 when a handful of silicon micro-strip detectors simply just ballooned the count. Growth in the sheer number of channels can be effectively absorbed by computing hardware up-scaling. As the growth mainly is due to finer segmentation of the detectors, the number of hit channels per event does not increase as much. Scaling-problems are rather related to the increased number of associated calibration parameters which must be handled — and more importantly — determined.

Detectors with different design and operating principles require different methods for calibration and reconstruction. Figure 11.2 shows the increase in

[1] The issue is not simple copying, but the ability to handle significantly larger and more complex problems, without letting the amount of code explode.

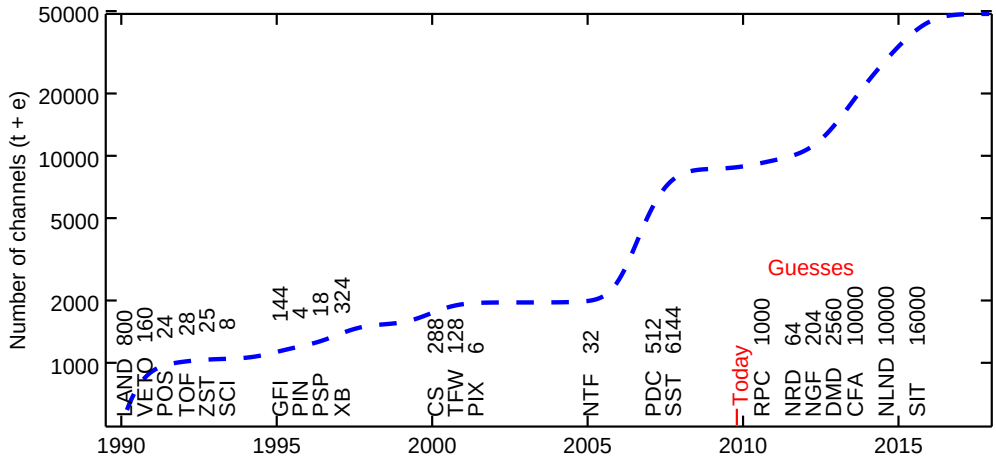

Figure 11.1: Channel count in the LAND setup since 1990. Each digitised channel (time or energy) is counted separately. The number of channels and commissioning years of future detectors under development and foreseen are educated guesses.

number of types of detectors which must all be handled by the software. The entries of Figure 11.1 which are not shown in Figure 11.2 represent devices with known and handled designs.

The computational growing pains associated with new detector types are essentially software related and can only marginally be battled with (computing) hardware. The issues can be alleviated by using more advanced programming techniques: generic code, templates, compartmentalisation / object-orientation. They simplify code re-use and prevent duplication. These techniques, however, also lead to higher thresholds for understanding the

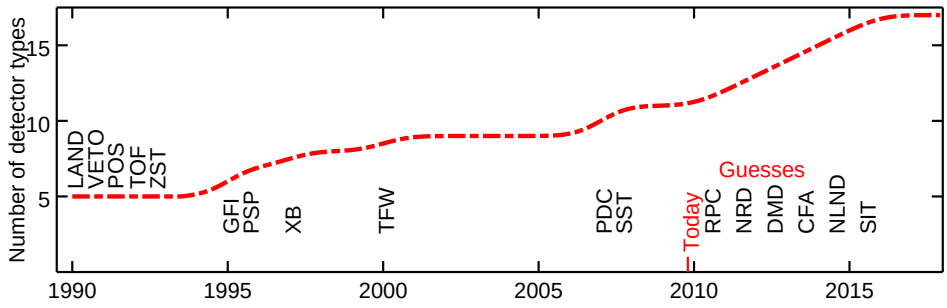

Figure 11.2: Detector types available in the LAND setup since 1990. Only designs known by `land02` and previously not used are counted.

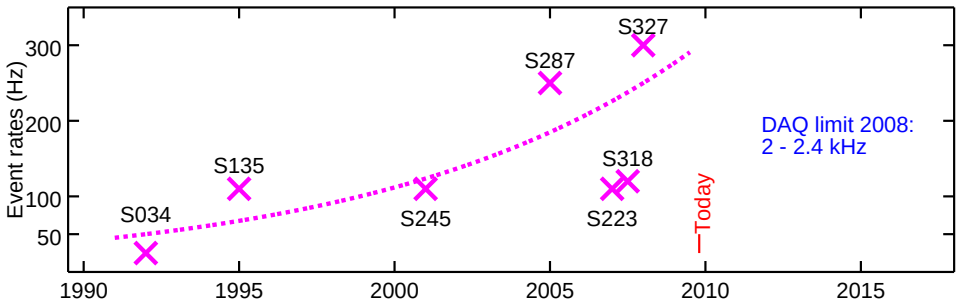

Figure 11.3: Average event-collection rates during physics runs in some LAND experiments, fitted with an exponential curve. Many experimental parameters beside the DAQ hardware affect the rates.

analysis software. Carefully applied, selected parts of the code can become simpler and easier to understand.

Figure 11.3 outlines the increase in event rates based on a few sample experiments. There are many factors affecting the amount of data collected in an experiment, only one of them is the intrinsic DAQ speed. The main factors are physics-related: the production rate of the radioactive isotopes under study, their reaction rates as well as triggering conditions. Increased amount of data is something which is easily accommodated by increased processing and storage capabilities. An important shift for analysis occurred in the early 2000s as hard drive capacities went beyond 100 GB and allowed easy and cheap access to the entire raw data sets of experiments. Tedious tape juggling is nowadays relegated to high-capacity tape-robots for long-term storage. By decreasing the turn-around times in analysis, more correlations can be carefully investigated.

The actual effective collection rates are only a fraction of what the DAQ speed limit would be at 100% dead-time. This limit was 2.4 kHz for the LAND setup in 2008. The margin is made up of several contributions: in order to trigger on interesting events, the trigger logics cannot issue triggers continuously, thereby limiting the amount of acceptable dead-time. The values reported in Figure 11.3 are average rates, and the duty cycle of the accelerator complex must also be accounted for. This is about 80% for LAND experiments with radioactive ions from SIS-FRS. The temporal micro-structure of the beam also affects the usable DAQ duty-cycle. Short periods of high ion flux cannot be used efficiently (eventually also suffering from pile-up) and periods with almost no ions are lost as idle time. Particularly the limitation on allowable dead-time can be lifted by using a trigger-less DAQ, allowing a

one-time step-like increase in the amount of events that can be collected.

11.2 To trigger or not to trigger

A classical event-based DAQ has a hardware trigger. Signals from many detectors are combined by logic operations to represent conditions that are likely to be associated with the occurrence of interesting events. Such notifications are sometimes subject to downscale, i.e. only a certain fraction are passed along, as they may not represent the most interesting kinds of events that are searched for in an experiment. Provided that the DAQ is not in dead-time, i.e. all DAMs are ready to handle a new event, the trigger request is accepted and the event is recorded. The rate of ions hitting a detector may be orders of magnitude larger than the number of interesting events that can be handled by a DAM, making some sort of read-out rate-limiting feature a critical part of an acquisition system.

The advantages of a hardware-triggered system are its relative simplicity and straightforwardness, the limited rate requirements on DAMs as only useful signals are converted, the fact that all DAM channels are live when the event comes, and that the data becomes directly organised as events. The drawbacks are the requirements of deriving coincidence signals from many detectors quickly to produce a selective trigger, the necessity to delay the signals to not reach the sampling points of the DAMs before the trigger decision has been drawn and distributed, and the global dead-time common for the entire setup, which poses a scaling problem as a setup grows. The need to quickly combine trigger information can be alleviated by using several trigger layers, the latter generally in some form of software, accepting or rejecting storage of already converted data. A special form of a second-level trigger is a fast clear system, which inhibits an already approved trigger shortly after distribution, aborting any commenced conversions.

The LAND setup uses a hardware trigger with downscale. One extension to a hardware triggered system is to make relatively simple but effective data-reduction cuts in a second layer software trigger after data has been converted and collected but before it is permanently recorded. One such example is the ESN setup at KVI (used in Paper III), which uses scintillators for triggering and fast readout electronics followed by DSP boards doing event selection based on proton track topologies in a few MWPCs.

The opposite, trigger-less, approach is to abolish the global dead-time and make every channel of a setup self-triggering. Each signal is recorded with a time-stamp, either it is the amplitude or the (high-precision) time-stamp itself that is of interest. The data streams are combined and time-sorted.

A total data-recording system would simply record all such collected data. Such an approach is useful for decay experiments, where the effects of one ion first passing through the setup and then decaying are separated in time and interleaved with the events of other ions. A setup using this technique is the GREAT spectrometer at JYFL [9].

Besides effectively inverting the drawbacks and advantages of a hardware triggered system, two issues are introduced by using a trigger-less approach: a high-precision time distribution system covering the entire setup is required. While being akin to the "master start" distribution needed in a triggered system, the stability is of greater concern as the reference signal no longer necessarily is distributed close in time to the signals of an event. Secondly, despite no longer having a global dead-time and thus also live-time and guaranteed availability of all converters at the occurrence of an accepted event, each converter has its own local dead-time after each previously handled and converted pulse. This can lead to difficulties of non-observation in the analysis, due to parts of the detectors being intermittently blind as a result of previous signals.

Recovering the advantages of a triggered approach entails two main operations: With all signals available as converted data, it is possible to run a software trigger / event selection routine before final data-recording which makes the same coincidence decisions that a hardware trigger would have done. The second advantage of a triggered system, where it can handle detectors with very high ion rates may translate to intractable data-rates in a trigger-less system. This would, however, mostly apply to a limited set of detectors, often located upstream the main setup, e.g. before a final exotic isotope separation stage. Those detectors are themselves generally not part of the trigger decision. By employing digital delay-lines, i.e. buffering the converted data locally, and receiving information about the time-ranges associated with ions actually reaching the main setup, the rates of data leaving the converter boards can be substantially reduced.

While it is possible to operate a setup in a combination mode, where parts of the setup run triggered and other operate trigger-less, such a practice is likely to attract the disadvantages of both regimes. Such a hybrid approach can serve as a path when gradually converting the electronics of a setup, as is suggested for the LAND $\rightarrow$ R^{3}B transition [67]. Designing and building detector electronics to be useful in both kinds of systems is advantageous and becoming easier, as modern internal designs of DAMs are often directly suitable for self-triggered or streaming operation. The main requirement to still be used in a triggered scheme is then the ability to deliver fast signals used as input to the global trigger needed by other conversion front-ends.

11.3 Transition issues

The R^{3}B setup for FAIR [67] will heavily rely on developments that are currently done at the existing setup. This requires that several systems that are presently running and used for experiments (also described in this thesis) can be used in the context of the R^{3}B setup. For the envisaged large channel counts of the new big detectors (calorimeter including silicon tracker and neutron detector) the question of building a common, low latency, hardware-trigger is likely to become an issue, requiring an alternative technique.

A trigger-less scheme as described in the previous Section requires that all front-ends support such a mode of operation, which is not the case for the existing hardware. In practise this means that the readout systems do not provide the required buffering capabilities, are not prepared to include a channel- or at least system-wide time-stamp, or simply lack (self-) triggering capabilities altogether. At least for the transition period, one therefore certainly has to live with a hybrid solution, allowing for a staged transition between the two regimes. This period can be expected to last for many years, such that it in the long run would be counter-productive to treat the related issues and solutions as temporary — however tempting that may be. In order to avoid losses in physics performance, this comes at the expense that the full system design has to be done with extended care. It also means that front-ends which are not supporting *any* of the two schemes of operation by e.g. relying on an external trigger signal but not providing one have to be enhanced or replaced during the transition.

From the data transport side the hybrid solution requires a transport and event building layer that can combine time-stamped triggered and purely channel driven data streams. As far as analysis tools are concerned, the aforementioned layer is expected to deliver an event-based data stream, which can be handled without breaking the existing analysis software. The necessary debugging tools to assure synchronisation between the event data streams and to ensure an overall efficient DAQ operation are also required.

11.4 Computational effort

In order to estimate the computing-related complexity of analysing an ALADiN-LAND experiment, the effects of channel counts, different detector types and event rates can be combined, e.g. by multiplication. Figure 11.4 compares this to computing hardware improvements illustrated by Moore's law, here with a doubling every two years [68]. Some remarks:

- The model of multiplying these three measurable quantities is just

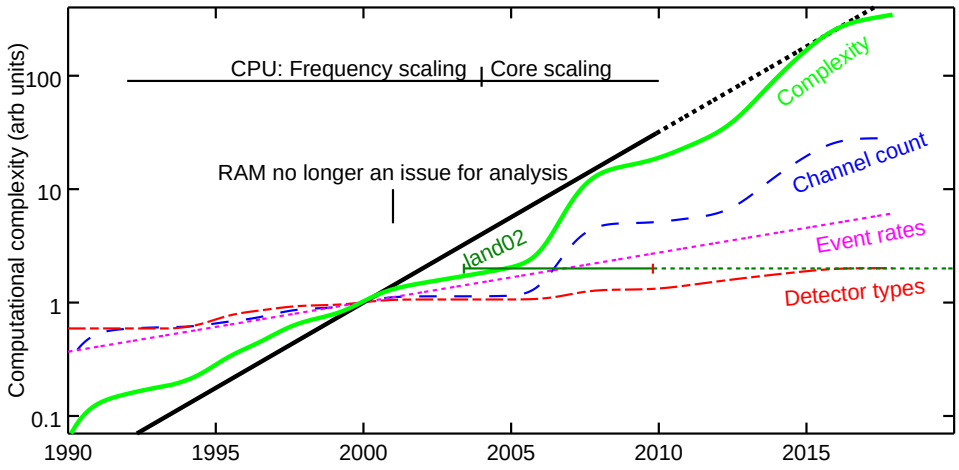

Figure 11.4: Computational complexity as estimated by multiplying the number of channels, detector types and event rates. The straight line represents Moore's law, here with a doubling every two years. The curves are somewhat arbitrarily normalised at the turn of the twentieth century. See the text for further discussion.

a crude and simplified estimate. Without defending the complexity-model itself, one may note that most obvious attempts at modifying it to lower the predictions for the future will require adjustments that at the same time increase the relative measure of the complexities that have been dealt with in the past. This is not realistic. A certain amount of the reduction in complexity with time is eaten up by the fact that more detailed studies can and are performed.

- The normalisation at the turn of the millennium (which could have been at any point) is not just devised for making a pretty picture. Also not only as it created the image of a period of *relative* ease of analysis from 2000 to 2005, which followed a reduced upgrade-rate. This bubble allowed `land02` to be initiated, take hold, and grow.

  The beginning of this decade has seen several shifts that affect analysis work:

  – The memory available even in modest desktops and laptops passed the point of being more than adequate for all event-wise data processing. Only special data collection during e.g. calibration may exhaust available memory and require extra implementation

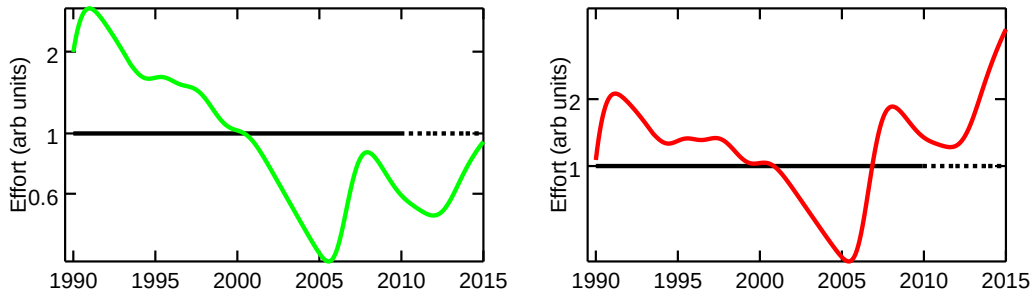

Figure 11.5: Computing effort (left) associated with analysing experimental data, as given by normalising the complexity, given in Figure 11.4, to readily available computing power, represented by Moore's law. The right picture is for a complexity model with an additional factor $\sqrt{\#\text{channels}}$.

care.

- Hard drive storage with capacity large enough to reasonably cheaply handle the raw data of entire experiments became readily available.
- Enhancements in processing capability has shifted from faster processors to more cores, i.e. more processing threads.

- Figure 11.5 (left) shows the computational complexity, inflation-adjusted for computing power, denoted as computational effort[2]. The surge in 2008, a result of the recent upgrades, has most certainly been felt by the affected experiments and data inquisitors — experimentalists.

To take advantage of the multi-core reality, software must be designed for parallel computations. This is non-trivial, but fortunately, computations for evaluation of nuclear physics experiments can be modified with moderate effort: being event-wise, and therefore easily partitioned, both event reconstruction and simulations lend themselves to parallelisation in relatively simple ways when compared to other kinds computational tasks: issue chunks of individual events to each computing thread.

Taking advantage of nowadays readily available compute architectures that take the multi-core concept to the extreme, e.g. GPUs or the CELL processor, is more difficult. Analysis procedures are built of short serial blocks with many branches, which is the very anti-thesis of routines suitable for

[2]The name *pain* sometimes feels more appropriate.

vector-processors. For efficient operation, these architectures require many threads of execution that follow the same execution path. Some computationally intensive hot-spots, e.g. tracking calculations that evaluate larger, more or less fixed formulas, should be possible to beneficially adapt. Extracting enough parallelism is, however, likely to require calculations for many events to be performed together, with increased demands on the overall scheduling of event processing.

11.5 Reality bites – tough medicine

The right panel of Figure 11.5 shows a slightly modified prediction, with the model amended by a factor $\sqrt{\#\text{channels}}$. This is to be viewed as the additional complexity introduced by having to deal with many more calibration parameters, and not with an increment in data volumes.

Improvements in computer hardware can handle the increase in data sizes. But unless the software(s) fulfil the requirements of the complex setup, no amount of computing hardware will help to extract the correct data. With evermore channels, each requiring the same care as today, capable calibration methods are needed. As is efficient reconstruction routines and easy handling of slow control parameters.

The software systems that are used, tested and developed today must be designed for tomorrow.

Many problems can be avoided; the cure is called preparations. Simulate experiments in advance. Not only with ion-optical calculations and projected rates of interesting events; exercise unpacking and calibrations and reconstruction — with realistic noise and pile-up rates and detector resolutions. The time from data-taking to results can then be reduced, by moving large fractions of the work to before an experiment. It especially applies to the introduction of new detectors and prototypes, which are especially prone to delaying analysis — when the associated software is developed post-experiment.

Glossary

ADC Amplitude-to-digital converter. Measures the maximum amplitude of detector signals during a gate. Commonly used for semiconductor detectors, e.g. silicon diodes.

ALADiN A large acceptance dipole magnet.

Borromean Three pieces interlocked such that when removing any of them, also the other two are set free. The name comes from the Bor-

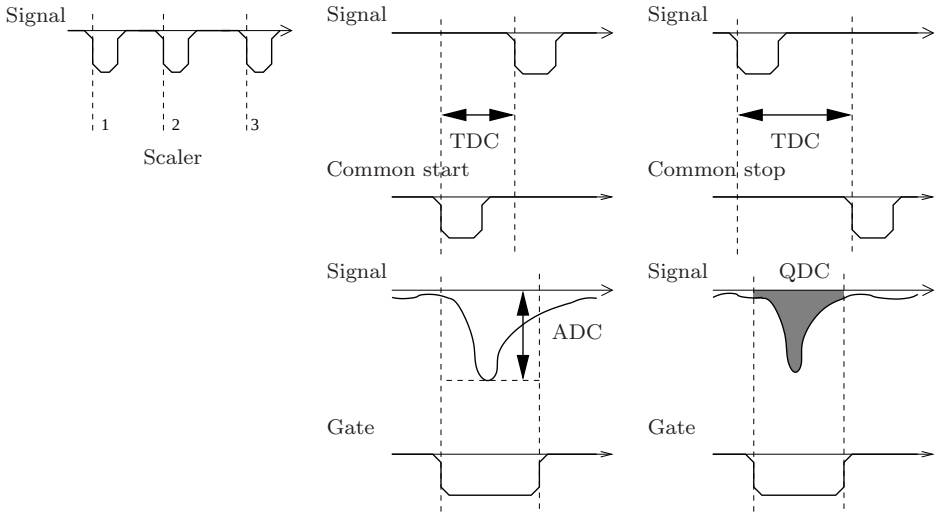

Figure 11.6: Basic DAM modules. From left to right on the first line: a scaler counting the number of logical signals and two TDCs operating in common start and common stop mode, measuring time interval between logical signals. On the second line, an ADC measuring the highest pulse amplitude (negative signal) during a gate, and a QDC integrating the signal during a gate.

romeo family, Italy.

CAMAC — Computer automated measurement and control. A data bus used frequently in nuclear physics experiments. Being replaced by the VME bus.

CERN — The acronym for its planning committee, Conseil Européen pour la Recherche Nucléaire, has stuck as the name for the world's largest particle physics laboratory, located outside Geneva, Switzerland.

CFD — Constant Fraction Discriminator. Determines the timing of a pulse, independently of the pulse height.

Channel — The word has two ambiguous usages. a) One detector or DAM channel = a read-out channel. b) The digital channels of one DAM channel (usually 4096), corresponding to the bins of a histogram of that read-out channel.

DAM — Digital acquisition module. See Figure 11.6 for examples.

DAQ — Data acquisition. The scoping of this word varies wildly — between only the program controlling the data read-out, to the entire read-out electronics, hardware as well as software.

Dead-time — Time during which another event cannot be accepted, due to one or more parts of the DAQ being busy, usually processing a previous event.

DLEB — Delayed event-building. Mode in which the MBS defers sending data over the network until off-spill pauses.

DSP — Digital Signal Processor. A kind of CPU which is particularly suited for repetitive processing of data (either integer or floating point, depending on processor model).

Endianess — Multi-byte words are stored in computer *memory* either with the most significant byte first (i.e. lowest memory address), or with the least significant byte first. The former convention is used by big-endian architectures, the latter by little-endian.

ESN — Eurosupernova, spectrometer detector setup at KVI, among others, measuring the (d,^{2}He) reaction on stable targets.

Event — The occurrence of each ion passing through a setup (possibly reacting), together with the data recorded from the detectors, is an event.

FASTBUS — A databus standard for nuclear physics experiments.

FERA — Fast encoding and readout ADC and TDC. Fast databus used with CAMAC modules to decouple single-event read-out from the slow CAMAC data bus.

FIFO	First-in first-out. A queueing principle.
FOPI	4π, experimental setup at GSI capable of detecting all charged particles emitted in heavy-ion reactions.
FRS	Fragment separator, magnetic spectrometer, located behind the SIS.
Gate	Signal determining when the inputs of a DAM module are open. Usually derived from the master start signal.
GSI	Gesellschaft für Schwerionenforschung.
HADES	High Acceptance Di-Electron Spectrometer, experimental setup at GSI investigating hadron properties inside nuclear matter.
HV	High voltage.
ISOL	Isotope Separator On-Line. Method for producing exotic isotopes by bombarding a heavy target, thereby fragmenting it and then extracting the isotopes by diffusion into an ion source. The low-energy beams of radioactive nuclei can be mass separated by a magnetic spectrometer.
ISOLDE	ISOL facility located at CERN, *on* the Swiss-French border, with the PS-booster bombarding the exotic isotope production targets with 1.4 GeV protons.
KVI	Kernfysisch Versneller Instituut, research laboratory situated north of Groningen, the Netherlands.
Lexer	Function that reads and interprets text input character by character, combining them into tokens, e.g. numbers, identifiers, keywords and strings.
MBS	Multi Branch System, a general DAQ framework actively developed at GSI.
Meta-data	Ancillary data describing an object. With a file, it could be its associated name, size, permissions and modification date.
MWPC	Multi-wire proportional chamber. Detector in which a gas is ionised by traversing charged particles. The charges drift and get amplified via repeated collisions by an electric field toward a grid of wires serving both as signal collectors and high voltage electrodes.
NIC	Network interface card. Computer subsystem responsible for handing network traffic.
NIM	Nuclear Instrument Module. A mechanical and electrical standard for the construction of experimental electronics.
Parser	Function that reads and interprets text input, generating an internal representation of the contents.

Pile-up Effect when a detector or other electronic component receives stimuli (hits or signals) so often that the output no longer is proportional to each input, but becomes a superposition (linear or non-linear).

PM tube Photo-multiplier tube. Detects photons (usually produced in a scintillator or crystals) by converting them into electrons and amplifies the electrical output signal by an avalanche of repeatedly multiplying the electrons.

QDC Charge-to-digital converter. Integrates and measures the current of detector signals during a gate. Commonly used for signals from PM tubes.

REX-ISOLDE Radioactive Beam EXperiment at ISOLDE. Post-accelerates exotic isotopes after production and separation.

RFIO Remote file I/O, protocol for handling high through-put large file access over the network, often accessing a front-end of a tape robot.

RICH Ring imaging Čerenkov detector. Uses the effect analogous to a sonic boom, whereby charged particles traversing a material faster than the effective speed of light generates a photonic shock wave. The photons are emitted in a cone, whose opening angle is determined by the particle velocity.

Round-robin Handling requests one after another, in a circular fashion.

RX/TX Receive and transmit. Terminology used in conjunction with network traffic.

S287 Experiment at the ALADiN-LAND setup in Cave C at GSI, April-May 2005, investigating the dipole strength in Ni isotopes.

Scaler Counts the number of pulses. Used to see how often a detector channel or other signal fires.

Scintillator Organic or inorganic compound which has a high probability of emitting photons (around the visible spectrum) after being excited by a passing ion.

SIS Schwerionensynchrotron. Accelerator at GSI.

Slow control Computer assisted adjustment and recording of the parameters controlling an experimental setup. Replacement for potentiometers and screw-drivers.

Spill Beams extracted from a synchrotron are delivered in long or short time periods, but with pauses. The pauses are needed for the synchrotron to accelerate the next load of ions. A spill is the time during which one load of ions is extracted towards an experiment.

TCP Transmission Control Protocol. Layer in a network communications protocol stack ensuring correct delivery of data.

TDC Time-to-digital converter. Measures the time between a common start or stop signal and individual detector signals.

TOF Time-of-flight. Used to determine the velocity of ions.

Trigger logic Part of the experimental electronics that, based on fast coincidences from the detectors, decide when an event has happened and if it should be recorded. It is also assuring the dead-time blocking.

VME VERSA module eurocard, a databus, commonly used in industry for computing and control applications.

VSB VME subsystem bus.

Walk Effect that distorts time measurements. Assuming that the shape of signals from a detector are the same irrespective of their amplitude, one would like to define the time of their occurrence uniquely, e.g. at the peak or when the signal amplitude has reached a certain fraction of the maximum value. When using a fixed threshold to generate the logical signal representing the signal time, the location will however depend on the signal amplitude. The dependence of the derived time signal on the amplitude (or shape) of the original signal is called walk.

`land02` Suite of programs to calibrate and reconstruct data collected with the ALADiN-LAND setup.

`ucesb` Unpack and check every single bit, program generator for unpackers of experimental data.

Acknowledgements

This work would not have been possible without all the enjoyable interactions with many people:

Thomas, the genuine interest in computing. With an environment for a physicist bitten by the programming bug.

Haik, letting me share your office for four years. It crucially defined this work. The continuous encouragement leading to more wild ideas.

Björn, Mikhail, Göran, encouragement, friendly advice and good suggestions. And the interesting stories!

A special thanks to Yuliya, covering for me by analysing S245 as I diverged into the data acquisition. Finding ways to get results, still never neglecting to do a thorough job.

Leonid, enthusiastically wringing physics out.

Christian, putting what we are doing into context.

Elisabeth, Mattias and Hannes, sharing offices on the fifth floor.

Nik, DAQ done right!

The probabilistic neutron trackers Johan, Hans and Linus.

The DAQ-team at the Uppsala proton test-beam: 2×Erik, Henry, Johan, Julius, and Martin.

Konstanze, discussing both this and that, with memories of early LAND.

Alina, Christine, 2×Christoph, Dominic, Felix, 2×Olga, Ralf, Tudi, Valerii, Günther, Marina, Michael, Rene, Tom — the LAND group, keeping me as a virtual member, so I never really left Germany. Your conscious and vigorous parallelised proof-reading in the moment of need was a great relief and most inspiring. Thank you!

Vasiliy, daring tester of new unpacker features.

Kripa, Stefanos, Audrey, who asked not what the programs could do for

them, but what they could do for future experiments.

Leonid G., encouraging conversations about data acquisition systems also being analysable.

Natalia, it was a great tour of Moscow!

Anders, Denis, Douglas, Horst, Jan-Eric, Jon, Lola, Marie, Olof, Roman, Saul, Sergei, Staffan, Stephane, Thomas, it was always a pleasure meeting you.

Pavel, next time we use a data inquisition system!

Kristian, letting me use your apartment.

Anton, nice chats in the corridor, with a smile.

Kate, cutting through administrativia.

Lars, dealing with obstructive hardware.

Pontus, Ulf and Andreas: Friday-pizza on Wednesdays.

The GSI computing department, Christo, Karin, Thomas, Stefan.

Henrik & Anna, welcome back to the front side of Sweden!

Evy & Lennart, you are simply the best!

Håkan Johansson, Göteborg, January 2010

Bibliography

[1] H. T. Johansson. *The DAQ always runs*. Lic. thesis, Chalmers University of Technology, Göteborg, 2006.
http://fy.chalmers.se/~f96hajo/lic/lic_thesis.pdf.

[2] M. Goeppert Mayer. On Closed Shells in Nuclei. *Phys. Rev.*, 74(3), 1948.

[3] M. Goeppert Mayer. Nuclear Configurations in the Spin-Orbit Coupling Model. I. Empirical Evidence. *Phys. Rev.*, 78(1), 1950.

[4] M. Goeppert Mayer. Nuclear Configurations in the Spin-Orbit Coupling Model. II. Theoretical Considerations. *Phys. Rev.*, 78(1), 1950.

[5] Yu. Aksyutina. *Light Unbound Nuclear Systems beyond the Dripline*. PhD thesis, Johann Wolfgang Goethe-Universität, Frankfurt am Main, 2009.
http://publikationen.ub.uni-frankfurt.de/volltexte/2009/7242/.

[6] M. V. Zhukov, B. V. Danilin, D. V. Fedorov, J. M. Bang, I. J. Thompson, and J. S. Vaagen. Bound State Properties of Borromean Halo Nuclei: ^{6}He and ^{11}Li. *Phys. Rep.*, 231(4):151–199, 1993.

[7] I. Tanihata, H. Hamagaki, Y. Hashimoto, O. and Shida, N. Yoshikawa, K. Sugimoto, T. Yamakawa, O. and Kobayashi, and N. Takahashi. Measurements of Interaction Cross Sections and Nuclear Radii in the Light p-Shell Region. *Phys. Rev. Lett.*, 55:2676, 1985.

[8] CBM Collaboration. Compressed Baryonic Matter Experiment — Technical Status Report, 2005.
https://www.gsi.de/documents/DOC-2005-Feb-447.html.

[9] I. H. Lazarus, D. E. Appelbe, P. A. Butler, P. J. Coleman-Smith, J. R. Cresswell, S. J. Freeman, R. D. Herzberg, I. Hibbert, D. T. Joss, S. C.

Letts, R. D. Page, V. F. E. Pucknell, P. H. Regan, J. Sampson, J. Simpson, J. Thornhill, and R. Wadsworth. The GREAT Triggerless Total Data Readout Method. *IEEE Trans. Nucl. Sci.*, 48(3), 2003.

[10] H.G. Essel and N. Kurz. The general purpose data acquisition system MBS. In *Real Time Conference, 1999. Santa Fe 1999. 11th IEEE NPSS*, pages 475–478, 1999.

[11] R. Hagedorn. *Relativistic kinematics: A Guide to the kinematic problems of high-energy physics*. Lecture Notes and Supplements in Physics. Benjamin, 1963.

[12] A. Winther and K. Alder. Relativistic Coulomb excitation. *Nucl. Phys. A*, 319:518–532, 1979.

[13] C. A. Bertulani and G. Baur. Relativistic Coulomb collisions and the virtual radiation spectrum. *Nucl. Phys. A*, 442:739–752, 1985.

[14] C. A. Bertulani and G. Baur. Electromagnetic processes in relativistic heavy ion collisions. *Phys. Rep.*, 163:300–408, 1988.

[15] G. R. Satchler. *Direct Nuclear Reactions*. Clarendon Press, Oxford, 1983.

[16] K. Kowalski, D. J. Dean, M. Hjorth-Jensen, T. Papenbrock, and P. Piecuch. Coupled Cluster Calculations of Ground and Excited States of Nuclei. *Phys. Rev. Lett.*, 92:132501, 2004.

[17] S. C. Pieper and R. B. Wiringa. Quantum Monte Carlo Calculations of Light Nuclei. *Annu. Rev. Nucl. Part. Sci.*, 51:53–90, 2001.

[18] P. Navratil, S. Quaglioni, I. Stetcu, and B. R Barrett. Recent developments in no-core shell-model calculations. *J. Phys. G*, 36:083101 (54pp), 2009.

[19] D. Habs, O. Kester, T. Sieber, H. Bongers, S. Emhofer, P. Reiter, P.G. Thirolf, G. Bollen, J. Äystö, O. Forstner, H. Ravn, T. Nilsson, M. Oinonen, H. Simon, J. Cederkäll, F. Ames, P. Schmidt, G. Huber, L. Liljeby, Ö. Skeppstedt, K.G. Rensfelt, F. Wenander, B. Jonson, G. Nyman, R. von Hahn, H. Podlech, R. Repnow, C. Gund, D. Schwalm, A. Schempp, K.-U. Kühnel, C. Welsch, U. Ratzinger, G. Walter, A. Huck, K. Kruglov, M. Huyse, P. Van den Bergh, P. Van Duppen, L. Weissman, A.C. Shotter, A.N. Ostrowski, T. Davinson, P.J. Woods, J. Cub, A. Richter, G. Schrieder, and the REX-ISOLDE collaboration. The REX-ISOLDE project. *Hyp. Int.*, 129:43–66, 2000.

[20] B. Rubio and T. Nilsson. NuSTAR. *Nuclear Physics News*, 16(1):9 – 14, 2006.

[21] D. H. Denby, P. A. DeYoung, T. Baumann, D. Bazin, E. Breitbach, J. Brown, N. Frank, A. Gade, C. C. Hall, J. Hinnefeld, C. R. Hoffman, R. Howes, R. A. Jenson, B. Luther, S. M. Mosby, C. W. Olson, W. A. Peters, A. Schiller, A. Spyrou, and M. Thoennessen. Ground state energy and width of ^{7}He from ^{8}Li proton knockout. *Phys. Rev. C*, 78(4):044303, 2008.

[22] W. R. Leo. *Techniques for Nuclear and Particle Physics Experiments*. Springer, Berlin, 1994.

[23] F. Humbert, T. Nilsson, W. Schwab, M. Zinser, Th. Blaich, M. J. G. Borge, L. V. Chulkov, Th. W. Elze, H. Emling, B. Franzke, H. Freiesleben, H. Geissel, K. Grimm, D. Guillemaud-Mueller, P. G. Hansen, R. Holzmann, H. Irnich, L. Johannsen, B. Jonson, J. G. Keller, O. Klepper, H. Klingler, J. V. Kratz, R. Kulessa, D. Lambrecht, Y. Leifels, A. Magel, M. Mohar, A. C. Mueller, G. Münzenberg, P. Møller, F. Nickel, G. Nyman, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, B. M. Sherrill, H. Simon, K. Stelzer, J. Stroth, O. Tengblad, W. Trautmann, E. Wajda, and E. Zude. Longitudinal and transverse momentum distributions of ^{9}Li fragments from break-up of ^{11}Li. *Phys. Lett. B*, 347(3-4):198–204, 1995.

[24] T. Nilsson, F. Humbert, W. Schwab, H. Simon, M.H. Smedberg, M. Zinser, T.H. Blaich, M.J.G. Borge, L.V. Chulkov, Th.W. Elze, H. Emling, H. Geissel, K. Grimm, D. Guillemaud-Mueller, P.G. Hansen, R. Holtzmann, H. Irnich, B. Jonson, J.G. Keller, H. Klinger, A.A. Korsheninnikov, J.V. Kratz, R. Kulessa, D. Lambrecht, Y. Leifels, A. Magel, M. Mohar, A.C. Mueller, G. Münzenberg, F. Nickel, G. Nyman, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, B.M. Sherrill, K. Stelzer, J. Stroth, O. Tengblad, W. Trautmann, E. Wajda, M.V. Zhukov, and E. Zude. ^{6}He and neutron momentum distributions from ^{8}He in nuclear break-up reactions at 240 MeV/u. *Nucl. Phys. A*, 598(3):418–434, 1996.

[25] M. Zinser, F. Humbert, T. Nilsson, W. Schwab, H. Simon, T. Aumann, M.J.G. Borge, L.V. Chulkov, J. Cub, Th.W. Elze, H. Emling, H. Geissel, D. Guillemaud-Mueller, P.G. Hansen, R. Holtzmann, H. Irnich, B. Jonson, J.V. Kratz, R. Kulessa, Y. Leifels, H. Lenske, A. Magel, M. Mohar, A.C. Mueller, G. Münzenberg, F. Nickel, G. Nyman, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, K. Stelzer, J. Stroth, A. Surowiec,

O. Tengblad, E. Wajda, and E. Zude. Invariant-Mass Spectroscopy of ^{10}Li and ^{11}Li. *Nucl. Phys. A*, 619(1-2):151–176, 1997.

[26] D. Aleksandrov, T. Aumann, L. Axelsson, T. Baumann M.J.G. Borge, L.V. Chulkov, J. Cub, W. Dostal, B. Eberlein, Th.W. Elze, H. Emling, H. Geissel, V.Z. Goldberg, M. Golovkov, A. Grünschloss, M. Hellström, J. Holeczek, R. Holzmann, B. Jonson, A.A. Korshenninikov, J.V. Kratz, G. Kraus, R. Kulessa, Y. Leifels, A. Leistenschneider, T. Leth, I. Mukha, G. Münzenberg, F. Nickel, T. Nilsson, G. Nyman, B. Petersen, M. Pfützner, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, W. Schwab, H. Simon, M.H. Smedberg, M. Steiner, A. Surowiec, T. Suzuki, O. Tengblad, and M.V. Zhukov. Invariant mass spectrum and α-n correlation function studied in the fragmentation of ^{6}He on a carbon target. *Nucl. Phys. A*, 633(2):234–246, 1998.

[27] K. Markenroth, M. Meister, B. Eberlein, D. Aleksandrov, T. Aumann, L. Axelsson, T. Baumann, M. J. G. Borge, L. V. Chulkov, W. Dostal, Th. W. Elze, H. Emling, H. Geissel, A. Grünschloß, M. Hellström, J. Holeczek, B. Jonson, J. V. Kratz, R. Kulessa, A. Leistenschneider, I. Mukha, G. Münzenberg, F. Nickel, T. Nilsson, G. Nyman, M. Pfützner, V. Pribora, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, H. Simon, J. Stroth, O. Tengblad, and M. V. Zhukov. ^{8}He-^{6}He: a comparative study of nuclear fragmentation reactions. *Nucl. Phys. A*, 679(3-4):462–480, 2001.

[28] M. Meister, K. Markenroth, D. Aleksandrov, T. Aumann, L. Axelsson, T. Baumann, M. J. G. Borge, L. V. Chulkov, W. Dostal, B. Eberlein, Th. W. Elze, H. Emling, C. Forssén, H. Geissel, M. Hellström, R. Holzmann, B. Jonson, J. V. Kratz, R. Kulessa, Y. Leifels, A. Leistenschneider, I. Mukha, G. Münzenberg, F. Nickel, T. Nilsson, G. Nyman, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, H. Simon, O. Tengblad, and M. V. Zhukov. Evidence for a New Low-Lying Resonance State in ^{7}He. *Phys. Rev. Lett.*, 88:102501, 2002.

[29] L. V. Chulkov, H. Simon, I. J. Thompson, T. Aumann, M. J. G. Borge, Th. W. Elze, H. Emling, H. Geissel, L. V. Grigorenko, M. Hellström, B. Jonson, J. W. V. Kratz, R. Kulessa, K. Markenroth, M. Meister, G. Münzenberg, F. Nickel, T. Nilsson, G. Nyman, V. Pribora, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, O. Tengblad, and M. V. Zhukov. Three-body correlations in electromagnetic dissociation of Borromean nuclei: The ^{6}He case. *Nucl. Phys. A*, 759(1-2):23–42, 2005.

[30] H. Geissel, P. Armbruster, K. H. Behr, A. Brünle, K. Burkard, M. Chen, H. Folger, B. Franczak, H. Keller, O. Klepper, B. Langenbeck, F. Nickel, E. Pfeng, M. Pfützner, E. Roeckl, K. Rykaczewski, I. Schall, D. Schardt, C. Scheidenberger, K. H. Schmidt, A. Schröter, T. Schwab, K. Sümmerer, M. Weber, G. Münzenberg, T. Brohm, H. G. Clerc, M. Fauerbach, J. J. Gaimard, A. Grewe, E. Hanelt, B. Knödler, M. Steiner, B. Voss, J. Weckenmann, C. Ziegler, A. Magel, H. Wollnik, J. P. Dufour, Y. Fujita, D. J. Vieira, and B. Sherrill. The GSI projectile fragment separator (FRS): a versatile magnetic system for relativistic heavy ions. *Nucl. Instr. and Meth. B*, 70(1-4):286 – 297, 1992.

[31] Th. Blaich, Th. W. Elze, H. Emling, H. Freiesleben, K. Grimm, W. Henning, R. Holzmann, G. Ickert, J. G. Keller, H. Klingler, W. Kneissl, R. König, R. Kulessa, J. V. Kratz, D. Lambrecht, J. S. Lange, Y. Leifels, E. Lubkiewicz, M. Proft, W. Prokopowicz, C. Schötter, R. Schmidt, H. Spies, K. Stelzer, J. Stroth, W. Walus, E. Wajda, H. J. Wollersheim, M. Zinser, and E. Zude. A large area detector for high-energy neutrons. *Nucl. Instr. and Meth. A*, 314(1):136–154, 1992.

[32] Yu. Aksyutina. private communication, 2010.

[33] R. C. Lemmon, O. Kiselev, and T. Nilsson. Quasifree hadronic scattering studies of exotic nuclei. GSI experiment proposal S296, 2004.

[34] M. Lantz, H. Johansson, and Yu. Aksyutina. Analysis of target position in the S245 experiment. Internal report, 2008.

[35] L. Charteris. *The Saint at the Thieves' Picnic*. Avon Publishing Co. Inc., New York, 1951.

[36] A.V. Belozerov, K. Borcha, Z. Dlougy, A.M. Kalinin, Nguyen Hoai Thiau, and Yu.E. Penionzhkevich. Study of properties of helium-isotopes in reactions with heavy ions. *Izv. Akad. Nauk SSSR, Ser. Fiz.*, 52(1):100–103, 1988.

[37] W. von Oertzen, H. G. Bohlen, B. Gebauer, M. von Lucke-Petsch, A. N. Ostrowski, Ch. Seyfert, Th. Stolla, M. Wilpert, Th. Wilpert, D. V. Alexandrov, A. A. Korsheninnikov, I. Mukha, A. A. Ogloblin, R. Kalpakchieva, Y. E. Penionzhkevich, S. Piskor, S. M. Grimes, and T. N. Massey. Nuclear structure studies of very neutron-rich isotopes of $^{7-10}$He, $^{9-11}$Li and $^{12-14}$Be via two-body reactions. *Nucl. Phys. A*, 588(1):c129 – c134, 1995.

[38] H. G. Bohlen, A. Blazevic, B. Gebauer, W. Von Oertzen, S. Thummerer, R. Kalpakchieva, S. M. Grimes, and T. N. Massey. Spectroscopy of exotic nuclei with multi-nucleon transfer reactions. *Prog. Part. Nucl. Phys.*, 42:17–26, 1999.

[39] K. K. Seth. Nuclear structure: Many faces, many probes. *Nucl. Phys. A*, 434:287–328, 1985.

[40] K. K. Seth, M. Artuso, D. Barlow, S. Iversen, M. Kaletka, H. Nann, B. Parker, and R. Soundranayagam. Exotic Nucleus Helium-9 and its Excited States. *Phys. Rev. Lett.*, 58(19), 1987.

[41] G. V. Rogachev, V. Z. Goldberg, J. J. Kolata, G. Chubarian, D. Aleksandrov, A. Fomichev, M. S. Golovkov, Yu. Ts. Oganessian, A. Rodin, B. Skorodumov, R. S. Slepnev, G. Ter-Akopian, W. H. Trzaska, and R. Wolski. T=5/2 states in ^{9}Li: Isobaric analog states of ^{9}He. *Phys. Rev. C*, 67(4), 2003.

[42] M. S. Golovkov, L. V. Grigorenko, A. S. Fomichev, A. V. Gorshkov, V. A. Gorshkov, S. A. Krupko, Yu. Ts. Oganessian, A. M. Rodin, S. I. Sidorchuk, R. S. Slepnev, S. V. Stepantsov, G. M. Ter-Akopian, R. Wolski, A. A. Korsheninnikov, E. Yu. Nikolskii, V. A. Kuzmin, B. G. Novatskii, D. N. Stepanov, P. Roussel-Chomaz, and W. Mittig. New insight into the low-energy ^{9}He spectrum. *Phys. Rev. C*, 76(2), 2007.

[43] Yu. Aksyutina, T. Aumann, K. Boretzky, M. J. G. Borge, A. Chatillon, L. V. Chulkov, D. Cortina-Gil, U. Datta Pramanik, H. Emling, C. Forssen, H. O. U. Fynbo, H. Geissel, G. Ickert, H. T. Johansson, B. Jonson, R. Kulessa, C. Langer, M. Lantz, T. Le Bleis, A. O. Lindahl, K. Mahata, G. Münzenberg, T. Nilsson, G. Nyman, S. Paschalis, W. Prokopowicz, R. Reifarth, A. Richter, K. Riisager, G. Schrieder, H. Simon, K. Sümmerer, O. Tengblad, H. Weick, and M. V. Zhukov. Heaviest helium and lithium isotopes. In *GSI Scientific Report 2008*, pages 154 – 155.
`http://www.gsi.de/informationen/wti/library/scientificreport2008/PAPERS/NUSTAR-EXPERIMENTS-08.pdf`.

[44] H. Simon, T. Aumann, M. J. G. Borge, L. V. Chulkov, Th. W. Elze, H. Emling, C. Forssén, H. Geissel, M. Hellström, B. Jonson, J. V. Kratz, R. Kulessa, Y. Leifels, K. Markenroth, M. Meister, G. Münzenberg, F. Nickel, T. Nilsson, G. Nyman, V. Pribora, A. Richter, K. Riisager, C. Scheidenberger, G. Schrieder, O. Tengblad, and M. V. Zhukov. Two-

and three-body correlations: breakup of halo nuclei. *Nucl. Phys. A*, 734:323–326, 2004.

[45] D. R. Tilley, C. M. Cheves, J. L. Godwin, G. M. Hale, H. M. Hofmann, J. H. Kelley, C. G. Sheu, and H. R. Weller. Energy levels of light nuclei A=5, 6, 7. *Nucl. Phys. A*, 708(1-2):3–163, 2002.

[46] R. Kanungo, C. Nociforo, A. Prochazka, T. Aumann, D. Boutin, D. Cortina-Gil, B. Davids, M. Diakaki, F. Farinon, H. Geissel, R. Gernhauser, J. Gerl, R. Janik, B. Jonson, B. Kindler, R. Knobel, R. Krücken, M. Lantz, H. Lenske, Y. Litvinov, B. Lommel, K. Mahata, P. Maierbeck, A. Musumarra, T. Nilsson, T. Otsuka, C. Perro, C. Scheidenberger, B. Sitar, P. Strmen, B. Sun, I. Szarka, I. Tanihata, Y. Utsuno, H. Weick, and M. Winkler. One-Neutron Removal Measurement Reveals ^{24}O as a New Doubly Magic Nucleus. *Phys. Rev. Lett.*, 102(15):152501, 2009.

[47] P. Haefner. Hinweise auf eine Soft-Dipole Resonanz im Halo-Kern ^{6}He. Master's thesis, Westfälische Wilhelms-Universität, Münster, 2004.
`http://www.uni-muenster.de/Physik.KP/frekers/publications/diplomarbeiten/dipl-haefner.pdf`.

[48] H. Törnqvist and L. Trulsson. Probabilistic Neutron Tracker. Master's thesis, Chalmers University of Technology and University of Gothenburg, Gothenburg, 2009.

[49] E. W. Dijkstra. How do we tell truths that might hurt? EWD498, 1975.
`http://www.cs.utexas.edu/users/EWD/ewd04xx/EWD498.PDF`.

[50] Physics Analysis Workstation.
`http://paw.web.cern.ch/paw/`.

[51] R. Brun and F. Rademakers. ROOT – An object oriented data analysis framework. *Nucl. Inst. and Meth. A*, 389(1-2):81 – 86, 1997.

[52] Y. Leifels. Analysis of LAND. Write-up.
`http://www-land.gsi.de/a_new_land/__public/documentation/analysis/analysis_of_land_YL/`.

[53] J. Cub, G. Stengel, A. Grünschloß, K. Boretzky, T. Aumann, W. Dostal, B. Eberlein, Th. W. Elze, H. Emling, G. Ickert, J. Holeczek, R. Holzmann, J. V. Kratz, R. Kulessa, Y. Leifels, H. Simon, K. Stelzer, J. Stroth, A. Surowiec, and E. Wajda. A large-area scintillating fibre

detector for relativistic heavy ions. *Nucl. Instr. and Meth. A*, 2(1):67–74, 1998.

[54] V. Blobel. Millepede II, Linear Least Squares Fits with a Large Number of Parameters. 2007.
http://www.desy.de/~blobel/mptalks.html.

[55] V. Blobel. Alignment Algorithms. In *Proceedings of the LHC Detector Alignment Workshop*. CERN, 2006.

[56] Phillips Scientific. 7120 Precision Charge Source / Time Interval Pulsar.
http://www.phillipsscientific.com/preview/7120pre.htm.

[57] J. R. R. Tolkien. *The Two Towers*. George Allen & Unwin, 1954.

[58] E. Tengborn. *Transfer reactions in inverse kinematics at REX-ISOLDE*. PhD thesis, Chalmers University of Technology, Göteborg, 2009.

[59] V. Blobel. Fast track-fit algorithm based on broken lines. 2005.
http://www.desy.de/~blobel/brline.html.

[60] FOPI A Detector for the Physics of Nuclear Reactions.
http://www-fopi.gsi.de/fopi_www_about.html.

[61] P. Adrich. Some notes about motion of the charged particle through the field of a dipole magnet and the mass measurement with the magnetic spectrometer ALADiN. Internal report, 2002.
http://www-land.gsi.de/a_new_land/__public/experiments/s221_2002/mass_res/matrix_motion.pdf.

[62] G. Audi, A. H. Wapstra, and C. Thibault. The 2003 atomic mass evaluation: (II). Tables, graphs and references. *Nucl. Phys. A*, 729(1):337 – 676, 2003. The 2003 NUBASE and Atomic Mass Evaluations.

[63] H. Geissel, C. Scheidenberger, P. Malzacher, and J. Kunzendorf. ATIMA.
http://www-linux.gsi.de/~weick/atima/.

[64] C. Sfienti. private communication, 2007.

[65] C.C. Paige and M.A. Saunders. Algorithm 583: LSQR: Sparse Linear Equations and Least Squares Problems. *ACM Trans. Math. Softw.*, 8(2):195–209, 1982.

[66] G. E. Moore. Cramming More Components onto Integrated Circuits. *Electronics Magazine*, 38(8), 1965.

[67] FAIR, An International Accelerator Facility for Beams of Ions and Antiprotons, Baseline Technical Report, Volume 4.5: R3B, 2006. `http://www.gsi.de/fair/reports/btr.html`.

[68] G. E. Moore. Progress in digital integrated electronics. In *Electron Devices Meeting*, 1975.

[69] V. Metag, R.D. Fischer, W. Kühn, R. Mühlhans, R. Novotny, D. Habs, U.v. Helmolt, H.W. Heyng, R. Kroth, D. Pelte, D. Schwalm, W. Hennerici, H.J. Hennrich, G. Himmele, E. Jaeschke, R. Repnow, W. Wahl, E. Adelberger, A. Lazzarini, R.S. Simon, R. Albrecht, and B. Kolb. Physics with 4π-γ-detectors. *Nucl. Phys. A*, 409:331 – 342, 1983.

Appendix A

Detectors of the LAND setup

This listing only contains detectors used in the experiments of the LAND group. Other experiments using ALADiN and/or LAND are not considered.

A.1 Past and present equipment

Entries in **bold** mark designs new or different enough to require or enable specialised calibration and reconstruction routines. Numbers in *italics* denote detectors that are no longer in use.

Name Channels	Tot. ch. # det	Description
LAND 400 t 400 e	800 1	20×10 paddles of sandwiched iron and plastic scintillator for conversion and detection energetic neutron. Each paddle is 2 m long and 10 cm wide and deep.
VETO 40 t 40 e	160 2	One or two planes of 20 scintillator paddles, one cm thick, covering the LAND front face to discard charged particles.
POS 4 t 4 e	24 3	Thin sheet ($\approx$ 300 μm) of scintillator used for time measurement of incoming ions. Originally also larger (20×20 cm^2) and thicker designs were used behind the target. The design with four PM-tubes allows for position determination, but requires considerable pin-cushion corrections.

Continues on the next page →

Name Channels		Tot. ch. # det	Description
WDC		*48*	Multi-wire proportional chambers, using a combination of delay-line read-out and charge division for coarse wire identification of the hit and drift-time versus an external reference for fine-grained interpolation of the hit position.
8 t	4 e	4	
CER		*8*	Čerenkov counter with radiator bars of glass for ion charge determination and triggering.
4 t	4 e	1	
TOF		*28*	Time-of-flight wall of a single plane of scintillating paddles for charged fragment detection.
14 t	14 e	1	
ZST		25	Multi-wire proportional chambers, with a delay-line read-out for position determination between closely spaced wires. The dual-stage amplification make them particularly useful for light particles with small charges.
5 t		5	
HAL		2	Plastic scintillator with annular hole, used as veto-detector upstream the target to discard wildly off-centre beam ions.
1 t	1 e	1	
MUS		*4*	Multi-sampling ionisation chamber for fragment charge determination with a volume of $40 \times 20 \times 20$ cm^3. The anode is divided into four 10 cm segments along the beam.
	4 e	1	
TDET		*132*	Target detector (γ) consisting of hexagonal BaF_2 crystals.
66 t	66 e	1	
NAJ		*22*	A few sodium-iodide scintillating crystal detectors used for γ detection.
1 t	1 e	11	
SCI		8	Single scintillator paddles used at the FRS focal planes to measure time and horizontal position.
2 t	2 e	2	
SIM		*16*	5.8×5.8 cm^2 silicon micro-strip detectors with 100 μm pitch and delay-line readout. Used for tracking of charged fragments behind the target.
8 t		2	
MUPL		*40*	Stack of five 34×9 cm scintillating paddles for multi-sampling of fragment time-of-flight and charge.
10 t	10 e	2	

Continues on the next page →

Name Channels		Tot. ch. # det	Description
GFI		144	50×50 cm^2 area of 500 vertical scintillating fibres for charged fragment position detection after deflection in the ALADiN magnetic field.
1 t	35 e	4	
KFI		*72*	Smaller GFI counterpart used close to the target. Replaced by PSP or ZST detectors.
1 t	35 e	2	
ROLU		16	Four motor-controlled scintillating blocks acting as veto for off-axis beam ions before the target. Named by the placement to the right, above, left and below the beam — rechts-oben-links-unten.
4 t	4 e	2	
PIN		4	Sheet ($\approx$ 300 μm) of silicon semiconductor diode operated with a reverse bias to deplete it of charge carriers. It only conducts current in pulses proportional to electron-hole pairs created by energy lost by passing ions.
1 t	1 e	2	
PSP		18	Position sensitive PIN-diode, realized by charge division between four corner read-outs. Effective for ions with $Z \gtrsim 10$.
1 t	5 e	3	
STR		*12*	Two pitch-fork scintillating detectors used in calibration runs to determine the PSP parameters. Replaced by PIX.
2 t	2 e	3	
XB		324	Darmstadt-Heidelberg Crystal Ball. 4π detector of 162 NaI segments used around the target for γ detection. Forward hemisphere recently upgraded with a second read-out chain of less amplification for recoil proton detection. This detector, 25 years [69] still going strong, predates the ALADiN-LAND setup — with the exception of ALADiN itself that came from CERN.
162 t	162 e	1	
CS		288	Alternative γ detector of 144 CsI crystals, only covering the forward hemisphere. Designed to fit closely around the target, it cannot be used with position sensitive charged recoil particle detectors surrounding the target.
144 t	144 e	1	

Continues on the next page →

Name Channels		Tot. ch. # det	Description
TFW		128	Dual-layer time-of-flight wall of 18 + 14, 10 cm wide, scintillating paddles for charged particle detection.
64 t	64 e	1	
PIX		6	Sheet of light-guide with $.5 \times .5$ mm^2 holes filled with scintillating material in a 2 mm spaced pattern. Used for position-calibration of the PSP detectors.
1 t	1 e	3	
FGR		30	Horizontally segmented plastic scintillator used at the FRS central focal plane for position and time measurement. Segmented to cope with high rates of particles, many of which will never reach the experimental target and setup.
15 t	15 e	1	
CV		32	Plastic scintillator bars used inside the CS γ detector to distinguish recoil protons.
16 t	16 e	1	
NTF		32	Yet another dual-layer time-of-flight wall of $8+8$ paddles, each 6.25 cm wide, with a total size to match the GFI acceptance.
16 t	16 e	1	
PDC		512	Drift chambers for detection of forward-emitted protons after deflection in the ALADiN magnet. Precise position is determined by drift-time differences to adjacent sense wires.
256 t		2	
SST		6144	Silicon micro-strip detectors for energy-loss and position detection of both fragments and light recoils. $640 + 384$ read-out strips of 100 μm pitch and charge-sharing gives sub-strip position resolution.
	1024 e	6	

A.2 Future devices – tentative

With the setup gradually becoming R^{3}B, many more new detectors are under development. They are to address issues of rate capability as well as more precise measurements. Without doubt, there is lots of work for `land02` to do, and it will have to learn new tricks.

Name Channels		Tot. ch. # det	Description
RPC		1000	Several layers of resistive plate-chambers are to be used for time-of-flight measurement of charged fragments. The use of RPCs will improve the time resolution.
500 t	500 e	1	
NRD		64	A low-energy neutron recoil detector, adapted to the needs of suppressing detection cross-talk due to inevitable multiple scattering of neutrons is under construction.
32 t	32 e	1	
NGF		204	It is being considered to build a new set of GFI-like detectors. The ease of use, including the new software for handling the intermediate analysis steps induced by the PSPM, make them attractive.
2 t	66 e	3	
DMD		2560	Beam-tracking in the SuperFRS requires detectors which can withstand and handle high intensities. Segmented diamond detectors are a candidate which can provide good time measurements.
256 t		10	
CFA		10000	γs emitted from relativistic ions require accurate Doppler corrections. The highly segmented target-detector CALIFA which also shall detect recoil protons, is being developed as two parts: a barrel and a forward end-cap.
5000 t	5000 e	1	
NLND		10000	A new large area neutron detector for energetic forward-focused neutrons is required to match the improved characteristics of the setup.
5000 t	5000 e	1	
SIT		16000	Precise determination of emission angles of recoil- and quasi-elastically scattered protons require a finely segmented dual-layer silicon strip-detector inside CALIFA.
	1000 e	16	