



Implementation & design approach to an Active SONAR Data Processing chain

R. Taylor
Analyst

Algorithmic Development
Thales Underwater Systems, Cheadle, UK

COMMERCIAL-IN-CONFIDENCE

GEDAE applications in TUSL

Introduction

- **Background to TUSL use of GEDAE**
- **Project coverage**
- **Development**

GEDAE applications in TUSL

TUSL background

- **MATLAB simulink**
- **RIPPEN**
- **ESPADON**
 - **Assessing a design Environment for Rapid Prototyping**
 - **GEDAE & PTOLEMY**

GEDAE applications in TUSL

ESPADON

- **Objectives**
 - Evaluate and demonstrate the method and performance of the **ESPADON Design Process** using representative **Systematic Signal Processing Applications**
- **Rapid prototyping**
 - Applied to both a **Sonar and Radar Beamformer Application**
 - Implemented on **Mercury, Europro and SKY platforms**
 - independence methodology / target
 - **Application to incorporate an Adaptive Beamformer.**
 - All GEDAE primitives, where possible.

GEDAE applications in TUSL

TUSL background

- **MATLAB simulink**
- **RIPPEN**
- **ESPADON**
- **PTOLEMY**
- **GEDAE**
 - **Ease of use**
 - **Support & feedback excellent**
 - **Covers SP, DP, databases etc.**
 - **Supports both C & C++ preferred**
 - **Large Support library**

GEDAE applications in TUSL

GEDAE project Support

- Predominantly used within Algorithm Development
- Currently used as the design environment for the New Norwegian Frigate program.
 - Responsible for providing a reference model for the entire Data Processing Chain.
- Providing the design and potential code delivery for major Project Signal Processing. Current work is GEDAE implemented Passive SP chain.
- Project support work on multiple programs. Used as a testing and development environment. E.g. towed body, shape correction.
- Trials Analysis. Ideal for processing large amounts of trial data.
- Development of a real time processor.
- **SFR&D** work. Looking at such areas as supervisory systems. GEDAE found to be quite flexible in this area.

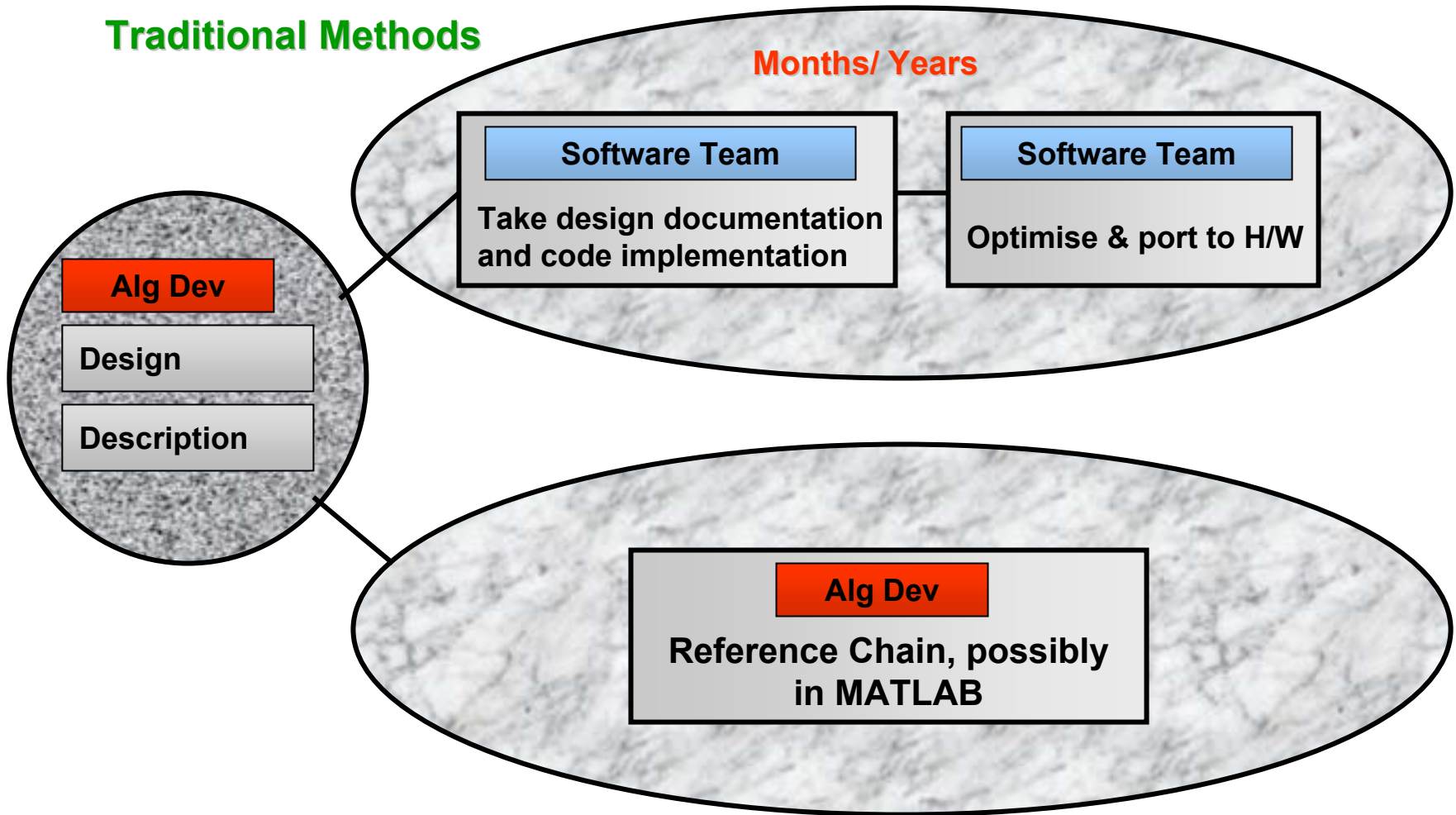
GEDAE applications in TUSL

Methodologies - Algorithm Development

- GEDAE predominantly sold as a SP development tool
- Primary proving Project is purely DP
- To save time in researching GEDAE primitives suitability for DP,
 - Source code written in C and harnessed in GEDAE primitives.
 - All bespoke Primitives
 - Some Historical C code
 - New C code development
 - C++ functionality.
 - GEDAE used to test and interface between different algorithmic modules.
 - Reference model held in GEDAE and tested against BL integrated H/W version.

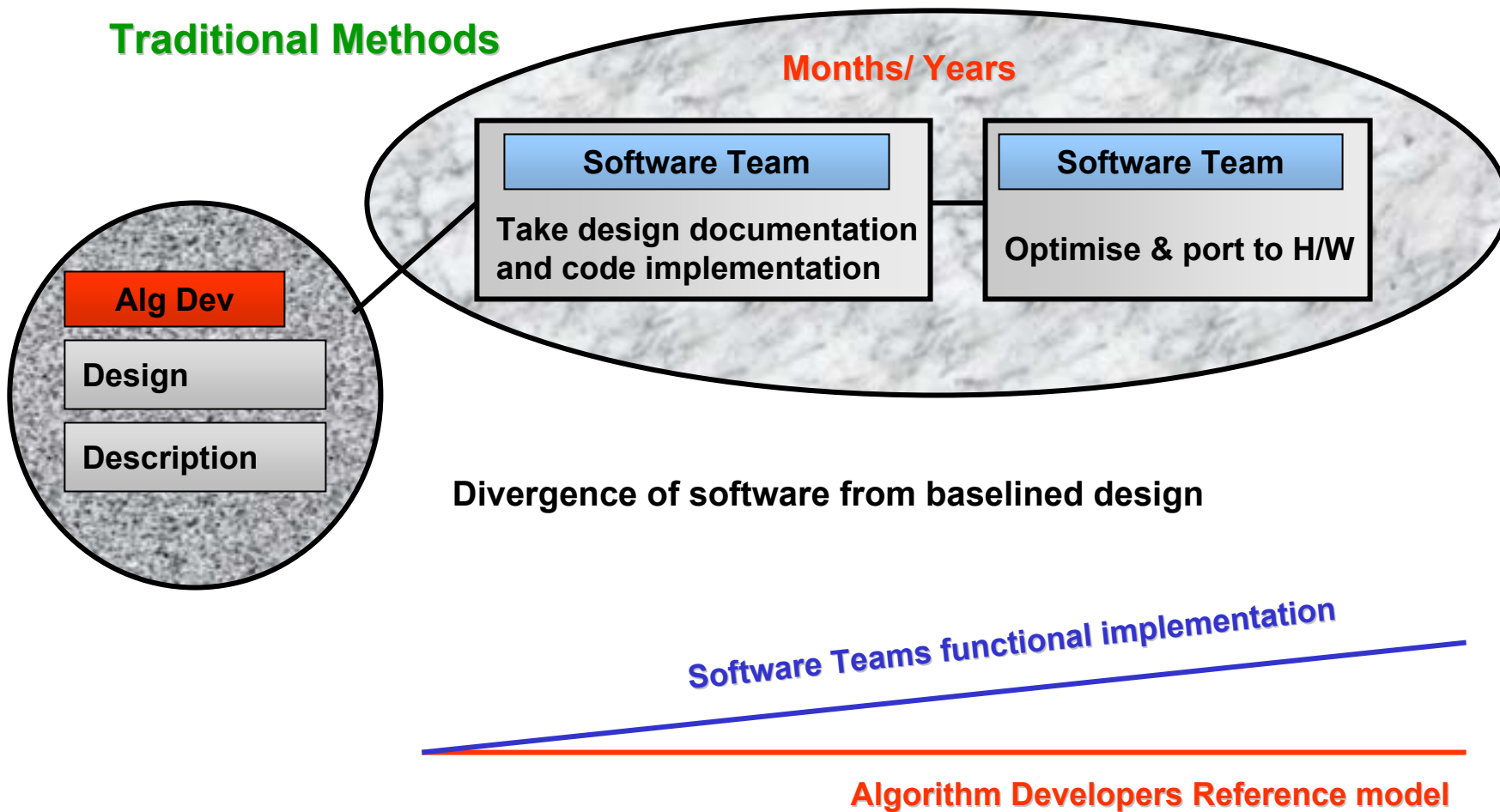
GEDAE applications in TUSL

Traditional Methods



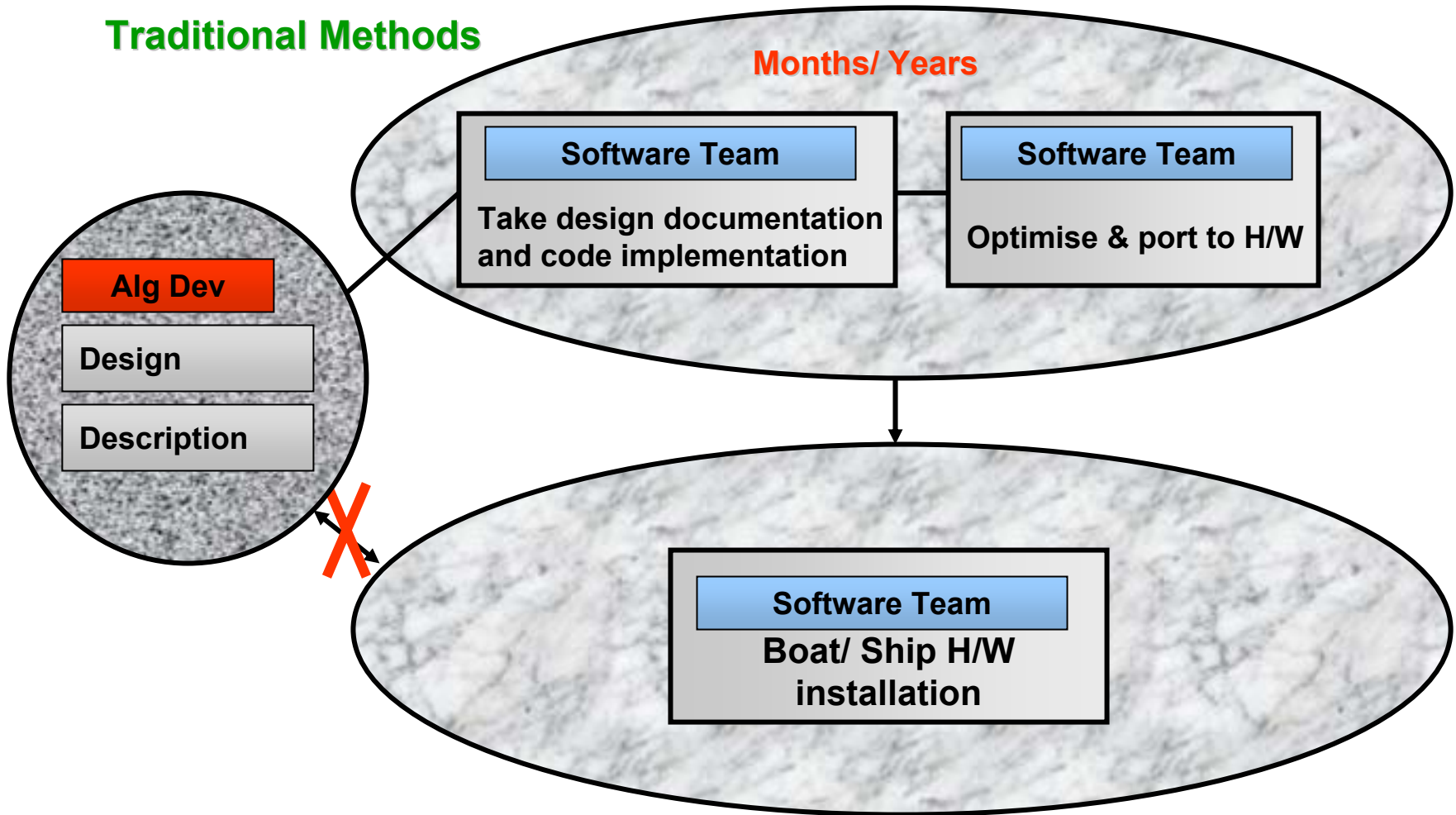
GEDAE applications in TUSL

Traditional Methods



GEDAE applications in TUSL

Traditional Methods



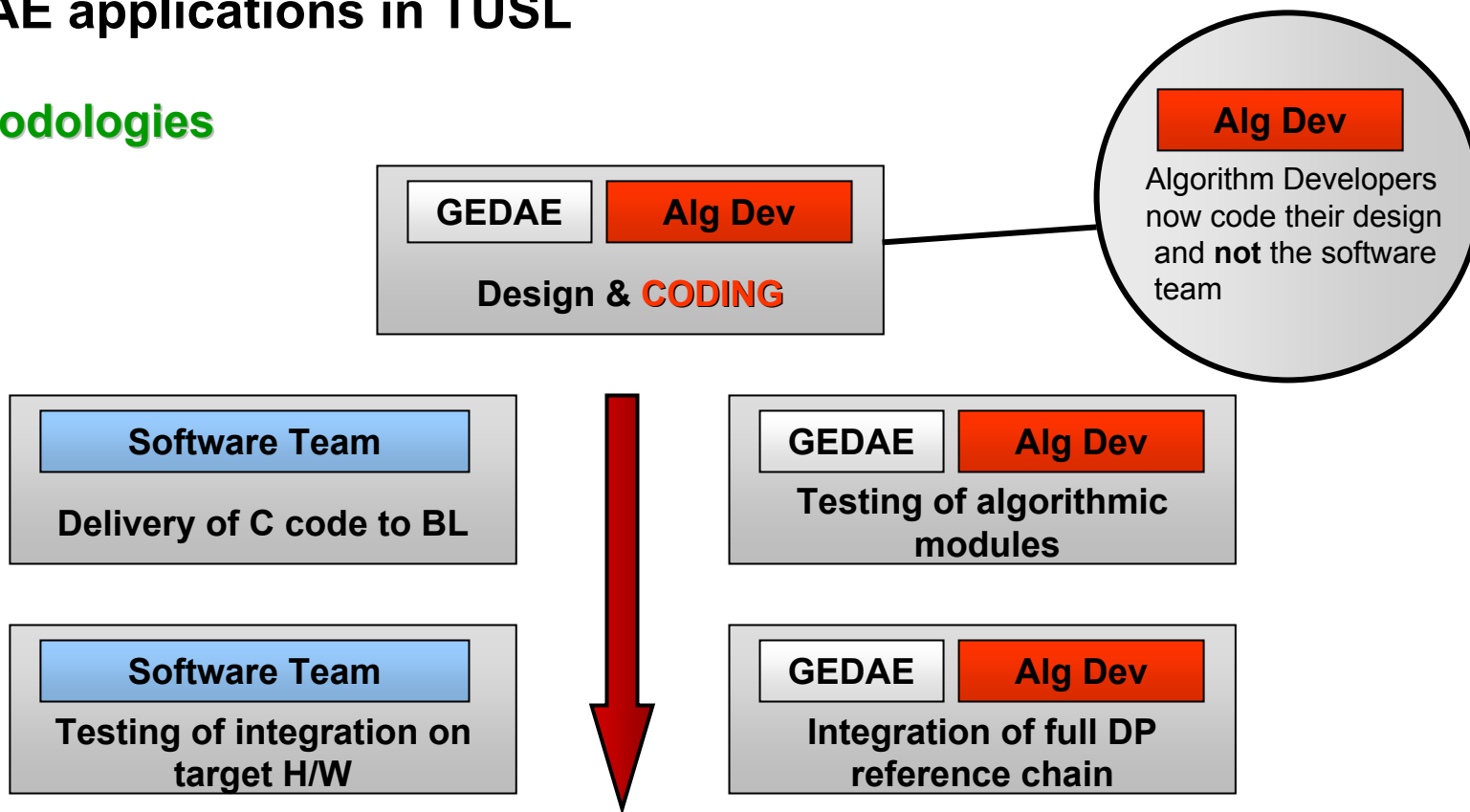
GEDAE applications in TUSL

Traditional Methods

- **Algorithmic Development team produce functional descriptions and reference chain.**
- **Software teams develop source code based on functional descriptions.**
- **Problems**
 - **Software team need a full understanding of the algorithmic functionality.**
 - **Larger projects can, over time, lose focus on original functional design and divergence between code and reference model occurs. A great deal of time can be spent correcting for this.**
 - **Changes to functional description require the software design process to start again.**
 - **Developers have no access to proving algorithms on H/W**

GEDAE applications in TUSL

Methodologies



Reference Model & Delivered Code are the same

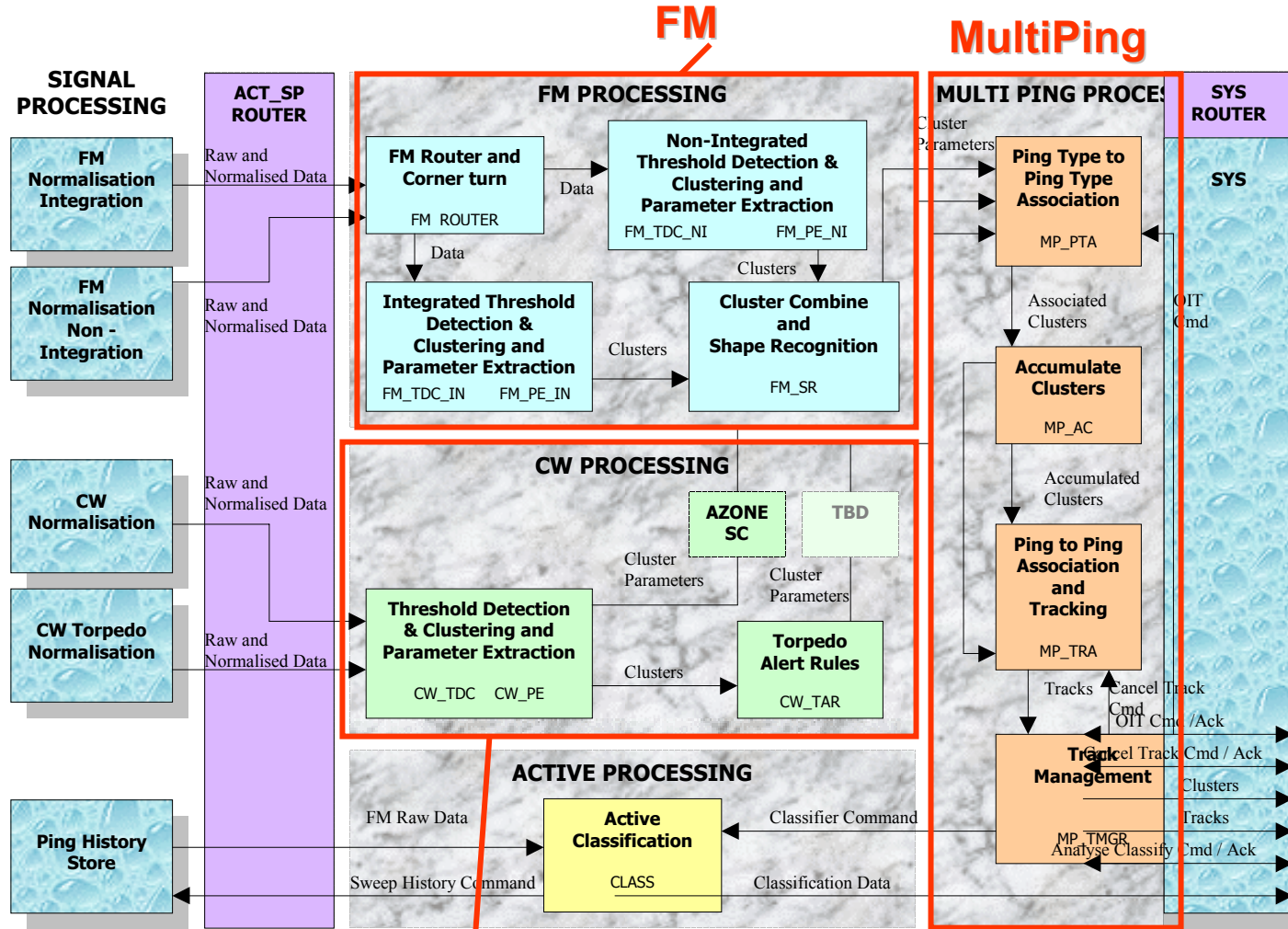
GEDAE applications in TUSL

Advantages to new methodology

- **Reference Model & Delivered Code are the same.**
 - Implementing same code on the target hardware by software team
 - Results on reference chain & target hardware should be identical.
 - Developers designs are now more mindful of hardware constraints.
- **GEDAE enables interface issues/ problems to be identified early.**
- **Testing & ‘tapping’ data from the processing stream done with ease.**
- **Primitives used by software teams to understand coding interfaces.**
- **Algorithmic team able to port designs onto H/W. Aids in loading estimates and facilitates rapid turn around for quick win trials projects.**

GEDAE applications in TUSL

Active Data Processing



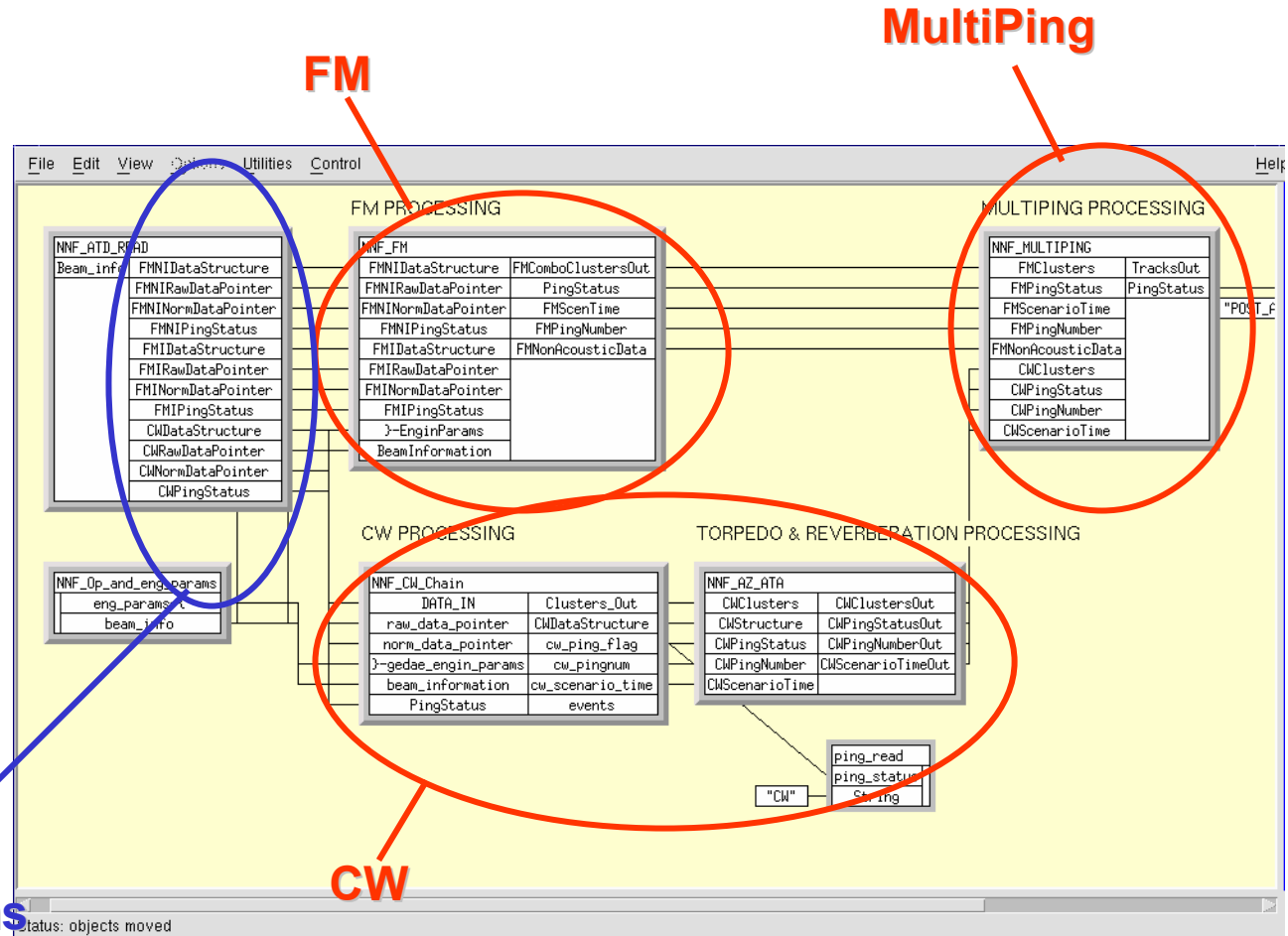
CW

GEDAE applications in TUSL

Methodologies

- 3 parallel processing streams - FM, integrated FM & CW
- Each stream processes data at different rates
- All primitives are bespoke.
- GEDAE enables us to use the **same** interface specifications as that defined by the **customer**.

Interfaces identical to customer definitions



GEDAE applications in TUSL

Data Processing

Module	Code
– Threshold, Clustering & Detection in FM & CW,	C
– Parameter Extraction in FM & CW,	C
– Cluster Associations in FM & CW,	C
– Shape Recognition in FM,	C
– Strong Contact removal application,	C
– Reverberation suppression in CW,	C
– Torpedo Alerts in CW,	C
– Ping to Ping Type Association,	C
– Multi- Ping Active Tracking,	C ++

GEDAE applications in TUSL

Scope of GEDAE influence in TUSL

- **Implemented Functionality, using GEDAE primitives**
 - Beamformer - with aid from BHDS this is almost running at real time.
 - Narrowband SP
 - Narrowband DP
 - Broadband SP
 - Broadband DP
 - DEMON
 - Noise estimation
 - **Active DP**
 - Ideal modular testing facility
- **SFR&D**
 - Algorithmic design and future technologies (see IDT talk)
 - Trial data analysis

GEDAE applications in TUSL

Signal Processing Projects

- **Currently Passive SP being designed in GEDAE**
 - Use of GEDAE only primitives
 - Current plans for code delivery, to be based on GEDAE auto generated code, providing access to GEDAE source is made available.
- **Active DP to be designed in a similar manner to previous Project again in GEDAE.**
 - Source code written in C and harnessed in GEDAE primitives.
 - GEDAE used to test and interface between different algorithmic modules.

GEDAE applications in TUSL

Other areas - BSPs

- **Unified Processing Architecture (UPA)**
 - **TUS chosen COTS processing environment**
 - PPC based processing, currently using Ceta boards and the LynxOS Operating System (Unix based)
 - Internal Middleware components
- **Decision to port GEDAE onto UPA**
 - **Initial port started mid-year, based upon existing port from BHDS (Unix port)**
 - Paul Bryant (Templecombe)
 - **Expected completion 3-4 months after start.**
 - Simple implementation based on standard unix port
 - Minimal use of optimised libraries
 - **Evolution options**
 - Integrate optimised libraries
 - Extend use of internal middleware components. (DDLlink, LCM, SCM, LOCO ...)

GEDAE applications in TUSL

Summary

- **GEDAE is TUSL preferred development environment for current and future projects.**
- **Within GSS , our 8 processor Sky box and BSP, enable us to quickly test new designs and ideas on real H/W.**
- **GEDAE used in support of both SP & DP processing + SFR&D.**

QUESTIONS?

