Introduction
The ELT era will mark the advent of a new generation of optical telescopes. In the past, advances in optical and mechanical technology have allowed for the progressive development of new technologies and tools, but have often overlooked the ability to systematically describe the set of associated (higher order) observatory functions and systems (at the outset).

With these considerations in mind ESO has started work on the control software for the ELT. The E-ELT is planned to have up to 10 complex software systems (including two post-atial optical systems) that will be built by external contractors (mainly new software) to which ESO will provide a system base and management of the complete system in operation. The E-ELT TCS is planned to have many of its subsystems built externally under subcontract. For all of these external contractors it is considered that there is need to develop a general framework in common in operation software is offered by design, but ESO will need to develop the frameworks that transform the software design to an operational use. This paper describes the framework for external contractors and collaborators in the process undertaken for analyzing and building the framework and described in this paper.

Software Frameworks
A software framework as discussed in this paper refers specifically to both the ELT Instrument Software (INS-WG) and the ELT TCS Infrastructure Framework (TCI). More generally, a software framework is “...a repository of reusable components that, in the form of a middleware, enables the implementation of an application through a template architecture...” (http://www.multimedia.org.au) in order to provide the potential usefulness of an INS-WG ESO considered it important to develop the template approach for new development of a comprehensive software framework, and the second-generation instruments for the VLT. This would provide a basis for determining the scope of any common software frameworks for the E-ELT project.

Creating development costs of different instruments is not easy. Development effort, design for commonality and component, cost and design for commonality and component, cost and design effort, are all needed. ESO has started work on such an analysis, but until now no common frameworks have been proposed. This should be the case, and one that can be mentioned and has seen that such framework is needed is that not only complexity and complicated frameworks can be easily measured.

For the time being, ESO has decided, in line with their expansion of instrumentation on the VLT both with and without software frameworks to be confident that the development of commonality and component design is not common. This approach of commonality and component design is not the same. Not using a common framework can mean that common problems for software frameworks and applications are solved with different efficiency (potentially costly errors) (e.g., leading to a system crash) and require different development effort.

On the other hand – using a framework could mean that a developer has to learn about the framework, and can lead to an increase in development time. An open question is whether or not supporting frameworks can provide support for specific systems provided in a generic way can be expensive.

Frameworks for the ELT Era – Requirements Analysis
The E-ELT Instrument Software (INS-WG) combines with specific external extensions to support each instrument, and includes all the software components and observatory functions and systems needed to package a new generation of complex software frameworks.

All E-ELT instruments currently being studied are unique, though some common functions will be incorporated. These new software frameworks must provide the possibility of a level of standardization. From the E-ELT observatory point of view, a single software framework should address commonalities, a general way that all the instruments are handled in a consistent, efficient and also flexible manner.

The challenge faced by both ESO developers, instrument and operations staff and external instrument developers is to maximize the efficiency and utility of the software framework. It will provide for the necessary standardization between the Observatories and the instrument but will not prevent the software from being unique to each observatory.

Also recognizing the complexity and individuality of each of the E-ELT instruments, the software framework that addresses a wide range of system functions. For example, this approach is a powerful instrument, as it will mean a resulting system that will be both complex and comprehensive. The ELT software framework will not just be the software framework for each instrument and also will be compatible with different systems.

The observable completeness of modeling and recording of individual data collections. The various functions and systems need to be able to be defined as a software framework. Therefore, this is developed from the basic understanding of what is to be provided and is managed.

The highest level goal is to:

- Provide a software infrastructure that can be used to control, connect and acquire data.

- It is expected that this will be done by:
  - An approach for developers to place together these components to form a system in an arbitrary way, but control structure that can be leveraged on a network.
  - An interface and communication infrastructure necessary for binding components of the system together.
  - A mechanism for the configuration of components.
  - A facility for extending the configuration of "specialized" components that support unique instrument requirements.
  - A mechanism for the independent command and control of individual components.
  - A mechanism for the observation of components and sequencing of actions between different components in the system.
  - A mechanism for the viewing and documenting of individual and collective data items (attributes) belonging to components of the system.
  - A mechanism in the display of images from scientific and wave front detectors.
  - A mechanism for the storage of both scientific and wave front image data using any INS-WG Data Product Specification which incorporates a database of metadata for each standard and non-standard image data.
  - A mechanism for the loading of data items and usage event systems for eventual storage in a database management system.
  - A mechanism to handle external systems events including the notification, acknowledgement, logging and sharing of alarms.
  - A mechanism to handle headers, error and exceptions.
  - A set of tools for real-time monitoring of telemetry data.
  - A mechanism to interface with the Cross-Repository Control System.
  - A mechanism to interface with the Observatory Astronomical Site Monitor to allow instruments access to providing astronomical and technological controls.
  - A mechanism to handle external systems events for eventual storage in a data base management system.
  - A mechanism to handle external systems events for eventual storage in a data base management system.
  - A mechanism to handle external systems events for eventual storage in a data base management system.

- It is a normal software architecture practice for any business rules applying to a system to be identified and stated at the outset. It is also important that all requirements stem from the product needs and rules and are constrained by these business rules. Business rules are derived from the specific business functions that the system (the INS-WG) is supposed to support.

- These rules have been derived from the E-ELT top-level requirements documents.