

4. APPROACH TO THE SERVICE

4.1 The Benefits of Reuse – An Introduction to Systematic Reuse

Purpose	The first course in a series covering a reuse-based approach to SPI. This course introduces the meaning of a reuse-based approach to SPI and helps you to define a reuse strategy adapted to the specifics of your company.
Description	With this two day course participants learn how to employ reuse within an organization with an emphasis on institutionalizing reuse practices to ensure that they are systematically employed in the company. The course demonstrates how a software process improvement programme can directly benefit the business objectives of a given organization.
Topics Covered	The course will address: the technical and organizational concepts of reuse, the impact of a reuse-based SPI programme on the software processes within a software development unit and throughout the organization, and the ingredients of a successful reuse-based SPI programme. The course will also demonstrate the benefits of reuse, including increased predictability in the software process, the opening of new business opportunities and a reduction of costs and time to market.
Target Audience	This course is suitable as an introduction to systematic software reuse. You should be considering systematic reuse as a means of improving the software development of these development activities. Participants should be a project leader, systems analyst or programmer, a software engineer or a technical manager, involved in software development and/or software process improvement initiatives.
Responsibilities	<p>ESI US</p> <ul style="list-style-type: none"> ▪ Provide full course material and an instructor <p>CLIENT</p> <ul style="list-style-type: none"> ▪ Provide adequate logistics for the course (training room, PC projector, flip chart, etc.)
Deliverables	<ol style="list-style-type: none"> 1. Students Manual including exercises and reference material. 2. Certificates of Attendance.
ESI Instructor	Iratxe Gomez - Instructor Profile in 6.1

4.2 Implementing Reuse with Flexible Components

Purpose	<p>This workshop shows how to implement reuse practices in a company using the approach of the RBSF (Reuse Based Software Factory). The emphasis of the workshop is in putting the reuse strategy into practice in an organizational context. This includes creating a set of reusable flexible components and establishing the conditions for their effective use.</p> <p>The course reports on experiences showing that, in less than half a year, an organization may obtain reuse rates of more than 60% using a limited set of 30-40 flexible components. Finally, course participants will have the opportunity to develop a mini-pilot reuse project.</p> <p>The course combines instructor presentations with hands-on exercises and uses a workshop style to promote participation and build on the participants' experiences.</p>
Description	<p>Flexible Components: A key technology for Software Reuse.</p> <p>A flexible component is a representation of a set of similar software components. Open decisions are explicitly captured in a flexible component by means of parameters of variation. Components instances are derived from the flexible component by binding the parameters of variation to valid values.</p> <p>You do not need to change programming language to enjoy the benefits of using flexible components. Flexible components technology is increasingly used in factory-oriented software development and since it makes no assumptions on your development or design languages, it may be used with absolutely any programming language, including COBOL, C, C++, Visual Basic, UML etc.</p> <p>Flexible components are at the core of the Reuse-based Software Factory (RBSF) approach, fostering software production in an industrial way to drastically reduce the cost of software production, exploiting similarities among developed applications.</p>
Topics Covered	<p>The course provides a comprehensive set of proven methods and techniques for the definition, design, implementation and usage of flexible components, including the realization of a mini-pilot to demonstrate their effectiveness.</p> <p>The participants will learn how to:</p> <p>*Increase reusability through standardization and</p>

Target Audience	<p>normalization</p> <ul style="list-style-type: none"> *Design flexible components to enforce commonality and to optimize on variable aspects *Construct programs as an assembly of flexible components *Deploy the approach in a company through a pilot project <p>Technical directors Software Department Managers Project Leaders responsible for teams of software developers Experienced analysts, designers and programmers (using any programming language) with the responsibility of implementing software systems, promoting a component-based, cost-effective and quality-driven approach</p>
Responsibilities	<p>ESI US</p> <ul style="list-style-type: none"> ▪ Provide full course material and the instructor <p>CLIENT</p> <ul style="list-style-type: none"> ▪ Provide adequate logistics for the course (training room, PC projector, flip chart, etc.)
Deliverables	<ol style="list-style-type: none"> 1. Students Manual including exercises and reference material. 2. Certificates of Attendance.
ESI Instructor	<p>Iratxe Gomez – Instructor Profile in 6.1</p>

4.3 Adopting MDD

Purpose	<p>This training will help you better understand MDD benefits, as well as how to approach this technology change in an efficient way. Over the last years Model Driven Development (MDD) has become an important software engineering strategy for handling complexity and the increasing requirements of larger and highly distributed software systems. This trend can be observed in domains as different as Telecommunications, Public Sector, Automotive and Air Traffic Management (ATM)</p> <p>The OMG’s Model Driven Architecture (MDA) has made, and continues to make, an important contribution to MDD through promoting industry supported standards such as the Unified Modeling Language (UML) and the Meta-Object Facility (MOF). MDA defines that, in a development process, there will be at least a platform independent and a platform specific model. This distinction has the advantage that</p>
----------------	---

	<p>platform changes resulting from technology progress can be addressed by model transformations meaning that the effort gone into problem specification can be reused wholesale to adapt the solution to the needs and requirements of the new platform.</p>
Description	<p>The idea of MDD is to use modeling languages as the fundamental pillar of which software development is based. Modeling languages, in the context of MDD can improve the productivity, quality, and longevity of software developments by reducing time to market, improving documentation, promoting reuse and helping to solve the problem of portability. The models, the central artifacts in the development, are composed at various levels of abstraction. Throughout the development life-cycle the level of abstraction is continuously reduced through the key concept of model transformation until the final, deployable system is created.</p>
Topics Covered	<p>This training course will train attendees on the new MDD methodology, through practical examples and based on proven processes, while learning how to deploy MDD at their organizations, how to use the proposed tools and what path to follow for the adoption MDD.</p>
Target Audience	<p>People with responsibility or direct involvement in the adoption of new technologies for the software development initiatives within their organizations, including:</p> <ul style="list-style-type: none"> • Intermediate Technical Managers • Software Process Improvement groups • High level senior analysts
Responsibilities	<p>ESI US</p> <ul style="list-style-type: none"> ▪ Provide full course material and an instructor <p>CLIENT</p> <ul style="list-style-type: none"> ▪ Provide adequate logistics for the course (training room, PC projector, flip chart, etc.)
Deliverables	<ol style="list-style-type: none"> 1. Students Manual including exercises and reference material.
ESI Instructor	<ol style="list-style-type: none"> 2. Certificates of Attendance. <p>Jason Xabier Mansell - Instructor Profile in 6.2 Asier Azaceta – Instructor Profile in 6.3</p>