

# **Effective eCommerce Development:**

## **Extending the Rational Unified Process (RUP) for eCommerce**

**Donald Firesmith  
Chief Methodologist (Architect)  
Lante Corporation  
DFiresmith@lante.com**

### Speaker:

- 15 years of experience developing object-oriented applications (software and systems).
- Author of 4 books, dozens of conference papers, and dozens of journal articles.
- Responsible for Lante's process and tools.
- A chief architect in the Technology Competency
- Based in the Dallas office

### Lante Corporation:

- Consulting/services company
- Develops eBusiness applications ("Nothing but Net")
- Concentrates on eMarkets
- Approximately 500 employees with extremely rapid growth.
- Multiple offices in US, Asia, and Western Europe (soon)
- Profitably In business for 15 years
- Recent Initial Public Offering (IPO) NASDAQ LNTE
- Partner with Rational (and many other companies)
- Lante has used the extensions listed in this paper on projects for numerous important clients including Dell, Airborne Express, BeautyJungle.com, SalesHound.com, Catepulse, and ZixIt.

## **Contents**

- **Lante's Customer Base**
- **What is RUP?**
- **RUP Weaknesses Become Lante Extensions**
- **Lante Process Framework**
- **Conclusion**

We will concentrate on the Lante Process Framework, which is an extension to the Rational Unified Process.

## **Lante's Customer Base**

- **Lante focuses on developing eMarket solutions for our customers:**
  - **.com start-ups**
  - **Click & Mortar**
- **Our Customers Maintain Systems**
- **Lante consists of Multiple Competencies**

Lante focuses on developing eMarket solutions for our customers:

- .com start-ups
- Click & Mortar

E-Market *systems* are typically:

- Large, complex, and business-critical.
- Provide a venue for multiple buyers and sellers.

Our customers typically require multiple eCommerce projects requiring an overall eStrategy.

Our customers maintain the systems we deliver:

- Systems must therefore be of high quality.
- They need complete, current, correct documentation

Lante consists of the following multiple competencies:

- Delivery Management (management, sales, and customer satisfaction)
- Technology (application development including testing)
- Strategy (develop of customer's overall eStrategy)
- User Experience (branding, look and feel style guide, etc.)

## What is RUP?

- **Rational Unified Process (RUP) is:**
  - **A de facto industry standard general purpose *software development method***
  - **A proprietary CASE tool**
  - **A great marketing tool for selling the Rational suite of tools**
- **We chose RUP for numerous reasons.**

RUP is:

- A de facto industry standard general purpose *software development* method.
- A proprietary process automation tool:
  - Tailorable HTML documentation of RUP
  - Website navigation tool
  - MS Word templates
- A great way to market the Rational tool suite (three Amigos, UML standard, lots of advertising)

We chose RUP as our baseline because:

- RUP is relatively complete, making it a good foundation on which to build (it is easier to tailor out what is not needed than to come up with missing stuff in the middle of a short schedule project)
- RUP's status as a de facto standard means that more of our recruits know about it and it is easier for our customers to accept it
- RUP promotes an iterative approach that our eMarkets customers need.
- RUP's phased approach allows us to manage risk for our customers and ourselves.
- RUP has wide tool support.

We modified RUP because of the reasons listed on the next slide.

## **RUP Weaknesses Become Lante Extensions**

- **Lante has extended RUP in the following ways:**
  - **Process Framework vs. Process**
  - **Delivery Cycle vs. Development Cycle**
  - **eMarket Focus vs. General Purpose**
  - **System vs. Software Focus**
  - **Requirements Engineering**
  - **Complete Conventions**
  - **Reuse and Component-Based Development**
  - **Additional Sources**

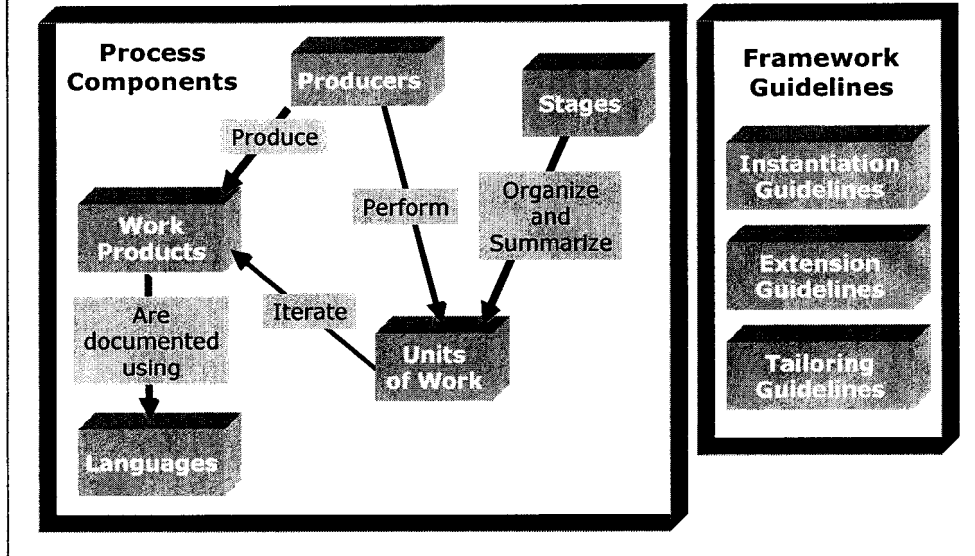
RUP does not currently adequately address:

- Delivery cycle issues
- Customer e-strategy of multiple projects
- Internet / WWW / User Experience issues (e.g., eBranding)
- Systems as opposed to software applications
- Quality requirements
- Conventions (procedures, guidelines, standards, templates, and inspection checklists)
- Reuse and component-based development

RUP does not adequately handle topics covered in other development methods:

- Responsibility Driven Design RDD (responsibilities, stereotypes, architecture guidelines)
- Design by Contract DBC (assertions and exceptions)
- Extreme Programming XP (automated up-front unit testing, refactoring, peer programming, user input on local estimation)
- OPEN Process Framework (more flexibility, better metamodel, more understandable terminology)

## Lante Process Frameworks



The Lante Process Framework (LPF) is Lante's proprietary framework for efficiently producing project-specific processes for delivering eCommerce/eMarket systems.

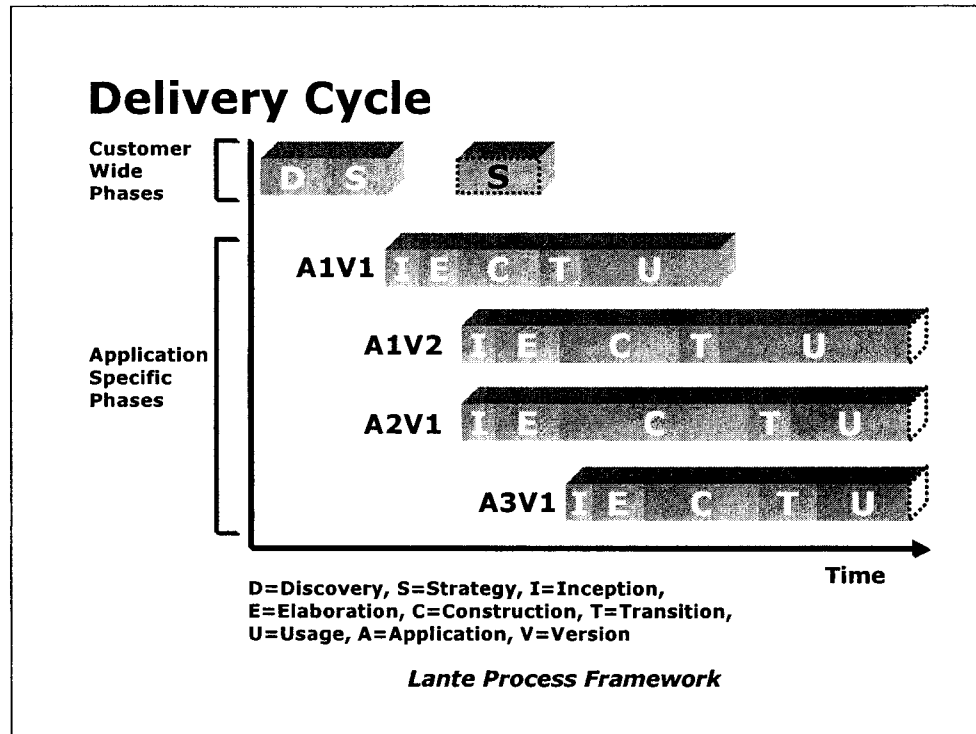
LPF is not a single method, because no method is flexible enough.

LPF consisting of:

- A class library of reusable, extensible, process components:
  - Work Products (Documentation, Software, Hardware, Applications)
  - Producers (Roles, Teams, Organizations, Tools)
  - Units of Work (Tasks, Activities, Workflows)
  - Stages (Cycles, Phases, Builds, Milestones)
  - Languages (Natural, Modeling, Implementation)
- A set of usage guidelines:
  - Instantiation guidelines (selecting components to make a project specific process)
  - Extension guidelines (No framework is so complete that new components will never be needed)
  - Tailoring guidelines (Once instantiated, you still need to tailor the components as the project progresses)

LPF must be instantiated to meet the specific needs of the project.

- May look like RUP++ if large, complex, business-critical, well-defined project.
- May look more like XP if small, simple, R&D, rapidly-evolving project.



RUP phases are in green; Lante added phases are in brown.

Discovery is pre-engagement marketing and sales.

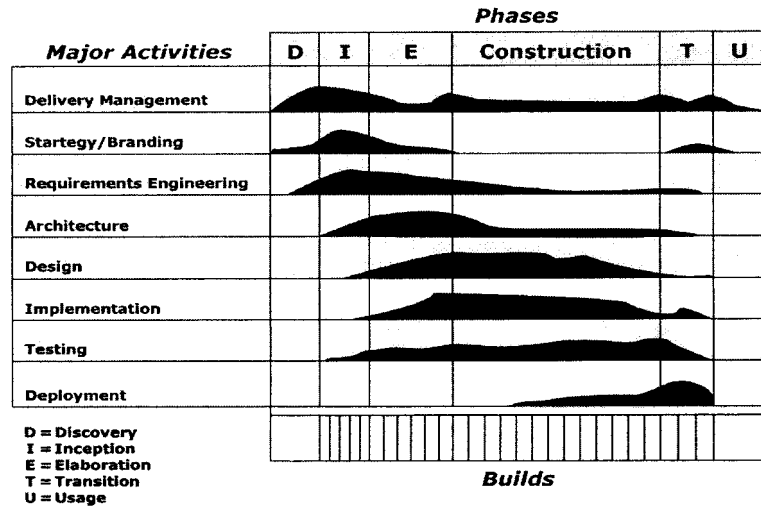
Strategy develops the overall client eStrategy and eBranding.

Strategy should (but does not always precede) the first inception phase.

Strategy and Inception are usually separate engagements.

Elaboration, Construction, and Transition are often combined into a single engagement.

# LPF Activities



RUP Business Modeling and Analysis have been merged into LPF Requirements Engineering which includes:

- Requirements Elicitation
- Requirements Reuse
- Requirements Analysis
- Requirements Specification
- Requirements Evaluation
- Requirements Management

RUP Design has been decomposed into LPF Architecture and Design.

RUP Project Management has been expanded into LPF Delivery Management.

RUP Workflows have become LPF Activities.

Discovery, Strategy, and Usage phases have been added.

Strategy and Branding activities have been added.



## **eMarket Focus**

- **Large, Complex, Business-Critical**
- **Internet and World Wide Web (WWW)**
- **Internet time (market-driven)**
- **Customer does not know most requirements**
- **Lante must be proactive**

LPF must support the development and delivery of large, complex, business-critical systems.

LPF must address:

- Internet and World Wide Web (WWW)
- Distributed Systems
- Browser/Internet technology
- User Experience (e.g., branding, look and feel, web site navigation)
- COTS Components (catalog management, content management, payment, fulfillment, personalization, ...)

Systems must be delivered in Internet time (market-driven):

- Cannot start from scratch (no time!)
- Must therefore reuse requirements, architecture, components, tests, etc.

Customer does not know most requirements:

- Especially .com startups
- Support requirements
- Quality requirements

Customer expects Lante to be proactive:

- Be eBusiness and eTechnology experts

## **Systems Focus**

- **Software-Intensive Systems**
- **Requirements:**
  - **System Requirements Specification**
  - **No Software Requirements Specification**
- **Architecture:**
  - **Systems Architecture Document**
  - **Software Architecture Document**
- **Testing:**
  - **Systems integration testing**
  - **Systems functional testing**
  - **Systems stress and load testing**
  - **Systems usability testing**

System consists of:

- Software
- Hardware (local and internet clients, networks, routers, firewalls, web servers, application servers, database servers, libraries, etc.)
- Wetware (system-internal personnel)
- Paperware (documentation, may be electronic content)

**System** Requirements Specification:

- System Operational and Quality Requirements
- System Design Constraints

No **Software** Requirements Specification!

**Systems** Architecture Document:

- Software, Hardware, and Personnel Components
- Relationships (Distribution), Responsibilities
- Buy vs. Build
- How architecturally significant requirements are met

**Software** Architecture Document:

- Software internal components (packages)
- Process and exception architecture

## Requirements Engineering

- Requirements Engineering
- Updated Use Case Modeling
- Emphasis on Quality Requirements
- Parallel with:
  - Customer visioning
  - Architecture
  - User Interface Prototyping

Requirements Engineering consists of:

- Requirements Elicitation (from customers and internal domain experts)
- Requirements Reuse (critical to meeting Internet time requirements)
- Requirements Analysis
- Requirements Specification
- Requirements Validation
- Requirements Management

LPF updates use case modeling:

- Adds Context Diagrams
- Added formality and additional information (e.g., business justification, textual requirements)
- Use Case Paths (scenarios)
- Risk Factors ala. John McGregor

LPF emphasizes the quality requirements, which hugely impact the system architecture.

The requirements engineering occurs during multiple phases and in parallel with customer visioning, architecture and design, architecture and user interface prototyping, implementation, and testing.

LPF produces:

- System Requirements Specification ( $\approx$  RUP Use Case Model + Supplemental Specification):
  - Use Case Model
  - Quality Requirements
  - Design Constraints

- Project Glossary
- API Specification ( $\approx$  RUP Supplemental Specification)
- GUI Prototype

## Quality Requirements

- RUP only weakly supports 4 quality requirements in the "supplementary specification"
- Quality requirements have a *huge* impact on the system architecture
- LFP supports:
  - **Auditability**
  - **Configurability**
  - **Correctness**
  - **Efficiency**
  - **Extensibility**
  - **Interoperability**
  - **Maintainability (RUP)**
  - **Operational Availability**
  - **Performance (RUP)**

**Auditability** (eCommerce)

**Configurability:**

- Functional Variants (quick and dirty to meet marketing window, quality)
- Internationalization
- Personalization

**Correctness:**

- Allowable Latent Defects
- Accuracy
- Precision
- Timeliness (currency and persistence)

**Efficiency**

**Extensibility** (rapidly changing and often unknown requirements)

**Interoperability** (with browsers, COTS components)

**Maintainability**

**Operational Availability** (global market implies 7x24)

**Performance:**

- Capacity
- Latency

- Response Time
- Throughput

## **More Quality Requirements**

- **Persistence** (how long)
- **Portability** (HW, OS)
- **Reliability (RUP)**
- **Reusability**
- **Robustness**
- **Safety**
- **Scalability**
- **Security**
- **Testability**
- **Usability (RUP)**

### **Portability**

Reliability

Reusability

Robustness

Safety

### **Scalability**

### **Security**

- **Identification, Authentication, and Authorization**
- **Content Protection**
- **Privacy (Confidentiality)**
- **Integrity**
- **Intrusion Detection**
- **Nonrepudiation**
- **System Maintenance**

Testability

Usability

- **Installation**
- **Usage (UX)**

## **Conventions**

- **Detailed Conventions:**

- **Coding and Content/Format Standards**
- **Templates**
- **Inspection Checklists**
- **Procedures (based on activities)**
- **Workflows (to produce work products)**
- **Guidelines**

- **Complete Example:**

- **C2C Auction and Sales eMarket (similar to eBay, uBid, etc.)**
- **Improves staff training in process**
- **Manages customer expectations**

RUP templates are overly simplistic and incomplete.

LPF provides detailed conventions:

- Coding and Content/Format Standards
- Templates
- Inspection Checklists
- Procedures (based on activities)
- Workflows (to produce work products)
- Guidelines
- Role, Team, and Organization descriptions

RUP examples are partial and not scalable to realistic examples.

LPF provides a complete, realistic example:

- C2C Auction and Sales eMarket (similar to eBay, uBid, etc.)
- Improves staff training in process
- Manages customer expectations

## **Reuse**

### **eTime & Quality imply Reuse of:**

- **Process (conventions)**
- **Requirements**
  - **Use Cases, Quality Requirements, Glossary, Domain Model**
- **Architectures:**
  - **Reference Architectures**
- **Designs and Frameworks**
- **COTS and home-grown Components**
- **Code**
- **Test Cases, Suites, and Drivers**

### eMarket

⇒ Large, complex, and business critical.

### eMarket and customer maintenance implies

⇒ High quality and well documented

⇒ Large effort

⇒ Long duration

### eCommerce

⇒ eTime

⇒ Short duration

### eMarket and Customer Maintenance and eCommerce implies

⇒ Apparent contradiction

⇒ False hidden assumption (development from scratch)

⇒ Massive reuse of all work products during all phases and activities.

Reuse helps support customer expectation that Lante be more proactive because Lante was selected as the eMarket expert.

## **Additional Sources**

- **Rational Unified Process (RUP)**
- **Lante eCommerce, eBusiness, and eMarket experience and expertise**
- **OPEN Process Framework**
- **Responsibility Driven Design (RDD)**
- **Design by Contract (DBC)**
- **Extreme Programming (XP)**
- **Patterns Community**
- **Standards Community**

In addition to being heavily based on the Rational Unified Process (RUP), the LPF is also based on:

- **Lante Experience**
  - Strategy (customer, market, user analysis; project identification, business case, selection), User Experience (branding, site navigation diagrams), Internet/WWW technology (browsers, XML, HTML, ...), hardware (routers, firewalls, networks, web servers, application servers, database servers, disk and tape libraries, etc.)
- **OPEN Process Framework**
  - Framework, Metamodel, Terminology, Teams and Organizations, multi-project programs
- **Responsibility Driven Design (RDD)**
  - Responsibilities, Stereotypes, Architecture Guidelines
- **Design by Contract (DBC)**
  - Assertions and Exceptions
- **Extreme Programming (XP)**
  - Unit Testing, Pair Programming, Planning Game (developer-set local schedules)
- **Patterns Community**
  - Patterns, Idioms, Frameworks
- **Standards Community**
  - Standards, templates, inspection checklists, examples



## **Conclusion**

- **RUP is incomplete for eMarket domain**
- **Much can and should be done to extend it**
- **The Lante Process Framework (LPF) extends RUP in many ways to support the development of high-quality eMarket systems**