- Analysis of commonality/conflict in functionality/architecture
- Reuse potential
- Evolutionary

**Social dimensions**
- Individual vs team vs organizational concerns
- Process — product — market congruence

*Implications for Process Technology*

**Generic Process**
- Parameterization, customization
- Fragments, composition
- Goals, motivations
- Concurrency, distribution
- Social considerations
- Practical (not research) modeling technology
  - Precision of the model
  - Specialization, generalization
  - Reflexivity, dynamism
  - Orthogonalization, interaction

**Process, Product Line Relationship**
- Process Line directed processes
- Mechanisms for Product Line (constraints) are reusable
- Congruency of process, product, and market
- Duality of process and product
- Specialize to the domain

*Process Technology Leverage*

**Rearrangement**
- Right time in the process (possibly just in time)

**Eliminate**
- Consider customer, business and no value
- Simplify
- Optimize

**Automate (that is, support enactment)**
- Agenda manager
- Workflow manager
- Animator
- Help, guidance
- Step/Activity automation
- Change propagation
- Cooperation and Coordination

*The utility of these comes from specialization.*
In this summary of product line implications for processes, I outline the major discussions and delineate their salient and relevant points. The summary presented below follows the viewgraphs presented in this session. The viewgraphs used in the summary are entitled:

- Techniques and Methods for Product Lines
- Product Line Aspects
- Generic Software Architecture
- Process Implications
- Implications for Process Technology
- Process Technology Leverage

**Techniques and Methods for Product Lines**

- Domain Analysis
- Software Architecture
- Platforms

- Generation
- Asset Reuse
- Customization
- Reverse Engineering/Rearchitecting

- Process Reuse
- Process Federation and Cooperation

- SW Configuration Management

**Product Line Aspects**

- Problem/Solution Space origin
- Shared assets
- System requirements - Business activity
- Similarity in some important aspect
- The result of narrowing, specialization
- Hyperweb of consistency constraints
- Trade-off decisions/strategic analyses
- Experience base
- Better, faster, cheaper

**Generic Software Architecture**

- Product line architectural drivers
- User requirements
- Domain constraints
- Business requirements
- Project constraints

- Instance ~ Type
- Harder ~ need better methods
- Good engineering even more critical
- Evolution is critical
  
  - Projection into the future
  - Instances ~ Type ~ Evolution

**Process Implications**

- Generation
  
  - Prototype oriented: specification ~ compile
  - Normal case first, then add special cases
  - Visual feedback useful
  - Tools for composition and analysis

- Asset generation (~ Product Line)
  
  - Asset base investment
  - Asset evolution
  - Asset base management
  - Asset use

- Asset Re-engineering (Products (~ Product Line)
  
  - Asset discovery
  - Asset synthesis and evolution
  - Continuous re-engineering

- Variability Management

- Change Management
  
  - Activity boundaries
  - Disjoint teams
  - Tool encapsulations
  - Isolation of workspaces
  - Multiple ‘streams’
  - product + processes

- Business domain
  
  - Domain analysis and bounding