5.1 Basic Technical Challenges

The producibility vision is achievable with the current abilities of competent developers. Techniques and supporting technology have gone through significant improvements over the years. Further advances in many aspects of software development practice would enable further improvements in the quality and effectiveness of software-based products. Improvements in support of challenging aspects of operational infrastructure would enhance productivity and quality for needed products.

This section describes important opportunities for improvement in two categories: development practices and the achievable behaviors of a product.

Developmental Opportunities

Developmental opportunities concern improvements in practices that developers use in building a product.

- Collectively revising and redeploying a set of similar deployed products, for both feature enhancement and defect diagnosis-repair, adjusting for differences resulting from each product's defining application model as well as customer-specific configuration settings and user-specific preference settings.
- Ensuring whole-product consistency and completeness are not degraded as products are modified (e.g., changes in deliverable elements (code and documentation) derive strictly from product-specific specification changes or due to product family-level modifications).
- All tools that are elements of a development platform support multi-versioned products and multi-product diversity multi-product tools (to simultaneously build and compare alternative solutions; rationale for diffs, which later changes overlap or conflict) (goal: understanding dependencies/rationale for results across tree of changes)
- Achieve prospective (prescriptive/predictive) quality versus only retrospective (descriptive/analytic) quality.

- Make tacit and heuristic (domain or circumstance dependent) knowledge explicit (implying some uncertainty and potential error) with any rationale, allowing it to be later questioned or refined (including viewing commonality assumptions as ground truth for a family).
- In modeling ecosystem reality, apply methods to ensure data integrity, currency, and consistency: account for the degree of uncertainty due to inaccessible/unavailable, missing/delayed, invalid, inaccurate, or imprecise measurements (using corrective approximation); correct for value divergence due to insufficiently current (aged) measures (interpolating from past values); maintain dependencies among values to avoid or correct inconsistent values (revaluation to replace dated or inconsistent values, using higher confidence or multiple information sources to infer better value). {info/data confidence, uncertainty/unknowns (costly to determine, unknowable, or falsified or suppressed due to secrecy/privacy/competitive risk)}
- data format impermanence (extensibility with computational efficiency given unforeseen need for info, subject content or metadata) (also see info overload) (current/historical/projected values: currently ad hoc problem-specific choice) (tradeoffs: storage/access efficiency, representational diversity, evolving schema for data and metadata, efficient for data values/relations/groupings)
- terminological flexibility (different terms/same meaning, same terms/different meaning - definitional context) (specifically, tailoring terminology to fit differing customer or market standards or conventions)

Operational Opportunities

Operational opportunities concern advances to enable developers to build products with more advanced capabilities.

(how to build software so as to account for these concerns; current: assume solutions are invariant/not-sensitive to them) (rationale: current practice is to ignore, solution may suggest method changes)

- failure/recovery, behavior prediction/rationale, degraded behavior under constraints
- Provide self-monitoring of behavior to detect or preclude any unspecified (out-of-scope) behavior (against what defn? related to explanation?)
- Support introspection-based explanation and rationale for behavior (dialog, logging, trail of cause-effect evidence).
- local/universal time, latency/variable comm-delay, transience/persistence (time is local, strategy/tasking can be across time but tactics/actions must be local; even "real-time" must account for latency; "instantaneous" communications is not actually, delay is inherent)
- info overload: data filtering & promotion (what data is relevant/needed & sent/stored to provide capab at progressive levels of processing) {for immed need vs for historic/retrospective/learning} {what to do with input from millions of petabyte env-sensor data streams: at each level, use? then discard or store/forward? ref latency}
- how to employ local, peer-to-peer, and multiple layers of server processing capab (every device is an encapsulated data service; some affect the env, directly or indirectly/via human/hdw) : not just which is initially cheaper: security, is info where it is needed when needed, how much delay is viable