2018
Measuring Sophistication in Systemic Design and Computing
Barba, Evan and Osborn, J.R.

Suggested citation:
Measuring Sophistication in Systemic Design and Computing

Evan Barba, J.R. Osborn

Communication, Culture and Technology
Program in Learning and Design
Department of Computer Science
Georgetown University, Washington, DC USA
Assumptions

Learning to integrate disciplines is at least as important as learning a discipline.

Interdisciplinary skills are *not* the same as disciplinary ones. But there is some overlap.

Objective metrics of learning are valuable.
Systems

Learning + Systems

Learning

Design + Learning

Design + Systems
This model can be applied to any three disciplines and intersections.
A Tripartite Seed
THE FUTURE
CREATING
CAPACITY OF
DESIGNING,
LEARNING
AND
SYSTEMS
GORDON ROWLAND

3 DISCIPLINES
&
3 INTERSECTIONS

More Disciplines & Intersections becomes unwieldy
Systems as a Way of Thinking
Design as a Way of Working
Computing as a Medium
The Next Generation Science Standards identify seven “cross-cutting concepts” that:

...need to be made explicit for students because they provide an organizational schema for interrelating knowledge from various science fields into a coherent and scientifically-based view of the world.

1. patterns
2. cause and effect
3. scale
4. system models
5. flows and cycles
6. structure and function relationships
7. stability and change.

“...new liberal art of technological culture,” (1992)

Richard Buchanan
Computing is where the job growth is.

1,000,000 more jobs than students by 2020

1.4 million computer jobs

400,000 computer science students

https://www.computerscienceonline.org/computer-engineering/
Learning Progressions

“Underlying any curriculum is a model of progression,”

M Hughes. 1996.

1. **Learning targets** that are defined by societal aspirations and analysis of the central concepts and themes *in a discipline*

2. **Progress variables** that identify the critical dimensions of understanding and skill that are being developed over time

3. **Levels of achievement** that define significant intermediate steps in conceptual/skill development

4. **Learning performances** which are indicative of skills and knowledge at each level, and which can be used in the development of assessments

5. **Assessments** that measure student understanding of the key concepts or practices and can track their progress over time.
Sophistication

“Learning is envisioned as a development of progressive sophistication in understanding and skills within a domain. [...] learning is conceived as a sequence or continuum of increasing expertise.”

Heritage 2008
Criticisms

No accounting for errors, failures, false starts…

Not interdisciplinary

Assumes all learners are alike
Eccentricity = 2.4 (depth of specialization)

Area = 41.25 (total sophistication)
Eccentricity = 2.4 (depth of specialization)

Area = 41.25 (total sophistication)
Advising

Image source: https://iris.peabody.vanderbilt.edu/module/tran-scp/cresource/q1/p01/
Table 1: Selection of levels defined for initial hypothesized learning progression

<table>
<thead>
<tr>
<th>Subject, Progress Variable, Assessment</th>
<th>Construct Map</th>
<th>Progress Guide</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Systems</strong></td>
<td></td>
<td>1. The system is defined, but critical components, the environment, and interactions are missing.</td>
</tr>
<tr>
<td>Progress Variable: System Mapping</td>
<td>1. Understands components and interactions and the distinction between system and environment; understands the basics of emergent and indirect effects.</td>
<td>2. All relevant components and interactions are present, stocks and flows are labeled, and feedback loops are noted as positive or negative and given qualitative character; the depiction is straightforward and understandable.</td>
</tr>
<tr>
<td>Learning Performance Category: System Map</td>
<td>2. Can elaborate the relationships between components (stocks and flows, feedback loops, etc.); can characterize and interrogate the interactions between the system and the environment.</td>
<td>3. Perturbations and interactions with the environment are noted and labeled and time is considered as a major factor and emergent effects are labeled, described or depicted.</td>
</tr>
<tr>
<td>3. Can predict how a system might adapt to changes and indirect effects</td>
<td>4. Scale and emergence are accounted for as are patterns of adaptation over time; the depiction is complex but parsimonious; multiple time-scales or perspectives might also be noted and depicted.</td>
<td></td>
</tr>
<tr>
<td>4. Can apply systems concepts in new contexts to find insight or explain phenomena</td>
<td>1. The user and system are described but the focus is on the immediate needs of the user; effects of the user on the system lack detail and do not unfold over time.</td>
<td></td>
</tr>
<tr>
<td><strong>Systems+Design</strong></td>
<td></td>
<td>2. The systemic constraints placed on the user (and the design) by the system are described in detail and the effects of the user on the system are clearly detailed.</td>
</tr>
<tr>
<td>Progress Variable: Intentional Emergence</td>
<td>1. Identifies primary component (typically a user) and understands its interaction with other components of a system.</td>
<td>3. The conceived solution is deployable and shows evidence of trade-offs needed to account for multiple perspectives.</td>
</tr>
<tr>
<td>Learning Performance Category: Field (Deployment) Study/Design Plan</td>
<td>2. Articulates indirect effects between the user and the system (i.e. constraints on the user imposed by the system and ways the user influences the system); adopts multiple perspectives in the design.</td>
<td>4. The observed behavior of deployed system is described in terms of both user and system effects.</td>
</tr>
<tr>
<td>3. Can identify trade-offs between User-Centered and System-Centered approaches</td>
<td>5. Iteration of solution makes appropriate trade-offs to optimize for both system and user.</td>
<td></td>
</tr>
<tr>
<td>4. Can identify emergent consequences of the intervention that affect both user and system</td>
<td>5. Can iterating to account for and optimize the observed emergent behaviors of both user and system.</td>
<td></td>
</tr>
<tr>
<td>5. Can iterate to account for and optimize the observed emergent behaviors of both user and system</td>
<td>1. Both the user and system are described but the focus is on the immediate needs of the user; effects of the user on the system lack detail and do not unfold over time.</td>
<td></td>
</tr>
<tr>
<td><strong>Design</strong></td>
<td></td>
<td>2. The systemic constraints placed on the user (and the design) by the system are described in detail and the effects of the user on the system are clearly detailed.</td>
</tr>
<tr>
<td>Progress Variable: Design Communication</td>
<td>1. Identifies opportunities for intervention and conceptualizes multiple solutions; likely gets fixated on one solution and cannot change course.</td>
<td>3. The conceived solution is deployable and shows evidence of trade-offs needed to account for multiple perspectives.</td>
</tr>
<tr>
<td>Learning Performance Category: Design Plans</td>
<td>2. Can create and follow a detailed plan resulting in a potentially deployable intervention; can communicate this plan at various points in multiple media; can adapt the solution partly.</td>
<td>4. The observed behavior of deployed system is described in terms of both user and system effects.</td>
</tr>
<tr>
<td>3. Can build a low-fidelity prototype; can deploy and revise the prototype based on user feedback and observation;</td>
<td>5. Iteration of solution makes appropriate trade-offs to optimize for both system and user.</td>
<td></td>
</tr>
<tr>
<td>4. Can add features that make the artifact robust to error and maintainable; documentation or user guides are clear; interventions and artifacts can persist</td>
<td>1. Both the user and system are described but the focus is on the immediate needs of the user; effects of the user on the system lack detail and do not unfold over time.</td>
<td></td>
</tr>
<tr>
<td>5. Can identify opportunities for intervention and conceptualizes multiple solutions; likely gets fixated on one solution and cannot change course.</td>
<td>2. The systemic constraints placed on the user (and the design) by the system are described in detail and the effects of the user on the system are clearly detailed.</td>
<td></td>
</tr>
<tr>
<td><strong>Design+Computing</strong></td>
<td></td>
<td>3. The conceived solution is deployable and shows evidence of trade-offs needed to account for multiple perspectives.</td>
</tr>
<tr>
<td>Progress Variable: Interaction Design</td>
<td>1. Can identify opportunities for interactive artifacts in a given context; can conceive of multiple assistive technologies and affordances of communication technologies that would be appropriate.</td>
<td>4. The observed behavior of deployed system is described in terms of both user and system effects.</td>
</tr>
<tr>
<td>Learning Performance Category: Prototyping</td>
<td>2. Can specify the technical requirements for a given design and understands the limitations of the technology in the context; can articulate the additional benefit adding technology would provide.</td>
<td>5. Iteration of solution makes appropriate trade-offs to optimize for both system and user.</td>
</tr>
<tr>
<td>3. Can build a low-fidelity prototype; can deploy and revise the prototype based on user feedback and observation;</td>
<td>1. Qualitative methods are used to understand a given environment and locate opportunities for design intervention. The proposed interventions are somewhat murky on details and not likely to be feasible due to poor understanding of the constraints.</td>
<td></td>
</tr>
<tr>
<td>4. Can add features that make the artifact robust to error and maintainable; documentation or user guides are clear; interventions and artifacts can persist</td>
<td>2. A feasible intervention is put forth that is tuned in to the needs of the situation and appropriately scoped; there is still little sense or plan to implement the solution or systematically test it.</td>
<td></td>
</tr>
<tr>
<td>5. Can identify opportunities for intervention and conceptualizes multiple solutions; likely gets fixated on one solution and cannot change course.</td>
<td>3. A simple prototype is created to prove the intervention along the lines of important features but this is used as the basis for iteration; there is a good sense of overall scope of the project.</td>
<td></td>
</tr>
<tr>
<td><strong>Computing</strong></td>
<td></td>
<td>4. Equal attention is paid to user expectations and technical implementation; quick iteration is seen as essential for success; features are removed rather than added to enhance stability and simplify the experience.</td>
</tr>
<tr>
<td>Progress Variable: Programming</td>
<td>1. Power User: can learn to use new tools, has an intuitive but naive sense for how data is represented and manipulated by these tools; can think through a problem in terms of logical steps and create a flow chart or similar representation.</td>
<td>1. Student can learn to use new tools easily, anticipates results and can combine sequences of actions to achieve desired results; has minimal understanding of how data is being represented and manipulated.</td>
</tr>
<tr>
<td>Learning Performance Category: Programs and exams</td>
<td>2. Understands the core elements of a computer program (syntax, control flow, variables, methods, debugging); writes pseudocode</td>
<td>2. Identifies and defines programming elements in a given program; can locate syntactical errors; describes algorithms and data representations using correct but loose terminology.</td>
</tr>
<tr>
<td>3. Can create a simple program in a well-defined problem context; can locate logical errors and debug syntactical issues</td>
<td>3. Simple program is functionally correct although spaghetti-like and overly complex; some features are likely not properly timed but the core is there.</td>
<td></td>
</tr>
<tr>
<td>4. Can implement a more complex program from a template; can formulate good questions when problems occur and seek out solutions from multiple sources but probably can’t determine which are most useful or debug logical errors</td>
<td>4. Successful implementation of a more complex program that uses reusable methods and incorporates available widgets among other advanced techniques with some scaffolding.</td>
<td></td>
</tr>
<tr>
<td>5. Can implement a complex program of their own design within a limited context; can locate external solutions and adapt them to their needs; can work with existing code bases, define new compund data types and integrate with outside services</td>
<td>5. Program incorporates techniques that were not taught; can ask well-formulated questions using technical terms correctly; Program is written outside of a sandboxed environment and makes use of professional grade tools, services, and software packages that the student identifies themselves.</td>
<td></td>
</tr>
<tr>
<td><strong>Computing+Systems</strong></td>
<td></td>
<td>1. Initial description has variables and relationships that are integral to the problem, but are far too complex to model realistically; a sense of questions the simulation could answer is demonstrated, but it is overly ambiguous and cannot provide a framework for the simulation to be meaningful.</td>
</tr>
<tr>
<td>Progress Variable: Modeling</td>
<td>1. Can identify variables appropriate for modeling in a given area of interest; Can organize verbal data and data by articulating relationships needed in the model.</td>
<td>2. Can implement a small-scale sim/viz by choosing appropriate tools; looks for trade-offs that need to made to make the model more robust; starts to ask questions about what the sim/viz might tell us that we don’t already know.</td>
</tr>
<tr>
<td>Learning Performance Category: Visualization/Simulation</td>
<td>2. Can implement a model of reasonable complexity and describe its features and limitations</td>
<td>3. A robust implementation is complete, and its limitations are being probed or articulated, revision of the initial questions is being considered; the sim/viz reveals some compelling behavior that can lead to insight.</td>
</tr>
<tr>
<td>3. Limitations of the implementation are clearly articulated; new insights can be drawn about the phenomena; and new extensions are conceived</td>
<td>4. The implementation is expanded to include features that reveal new things about the phenomena, such as analytics or interactive elements to reorganize the data.</td>
<td></td>
</tr>
</tbody>
</table>
Learning Progressions

“Underlying any curriculum is a model of progression,”
M Hughes. 1996.

1. Learning targets that are defined by societal aspirations and analysis of the central concepts and themes in a discipline

2. Progress variables that identify the critical dimensions of understanding and skill that are being developed over time

3. Levels of achievement that define significant intermediate steps in conceptual/skill development

4. Learning performances which are indicative of skills and knowledge at each level, and which can be used in the development of assessments

5. Assessments that measure student understanding of the key concepts or practices and can track their progress over time.
Questions