Assessment Engineering in *Test Design, Development, Assembly, and Scoring*

Richard M. Luecht, Ph.D.
University of North Carolina at Greensboro

07 November, 2008

East Coast Organization of Language Testers (ECOLT) Conference

Assessment Engineering (AE)

- AE provides an integrated framework with replicable, scalable solutions for assessment design, item writing, test assembly, and psychometrics
- Possible applications are being explored for multidimensional, K-12 classroom formative assessments
- Current applications are actually being developed for large-scale, summative assessment applications (e.g., the Uniform CPA Examination, AP, and the PSAT)

Assessment Engineering (AE)

- AE begins with the development of one or more **construct maps** that describe concrete, ***ordered*** performance expectations at various levels of a proposed ***scale***
- *Empirically driven* **evidence models** and **cognitive task models** are developed at specified levels of each construct, effectively replacing traditional test blueprints and related specifications
- Multiple assessment **task templates** are ***engineered*** for each task model to control item difficulty, covariance, and residual errors of measurement
- Psychometric procedures are used as **statistical quality assurance mechanisms** that can ***directly*** and ***tangibly*** hold item writers and test developers accountable for adhering to the intended test design

Why is AE Useful? Necessary?

- Psychometric models are "***data hungry***"
  - *Sparse data* is a serious problem for IRT and other psychometric models re **calibration**
  - AE can **reduce item exposure risks** by expanding item banks in a principled way
  - AE assessments capitalize on replication to **reduce item production costs** and overall pretesting costs
- Strong, empirically based **quality control** (QC) mechanisms can be implemented to improve test development in a concrete way
- AE is fully consistent with advanced psychometric models for calibration, equating, and scaling (e.g., hierarchical Bayes estimation and so-called *cognitive diagnostic models* and related constrained latent class models)

Recent Developments

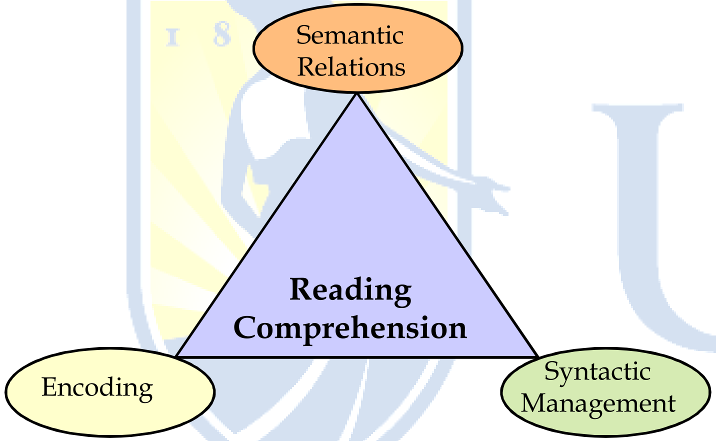
- Task design frameworks are making progress
  - Evidence-centered design (ECD, Mislevy & Almond)
  - Integrated test design, development, and delivery (ITD^3, Luecht)
  - AE design of accounting simulations (Luecht, Gierl, and Devore, 2007; Luecht, Burke, & Devore, 2008)
  - Language testing (Kenyon, 2007; Tucker, 2008)
- Automated test assembly is in place (van der Linden, 1989, 1998, 2005; Luecht, 1992, 1994, 1998, 2000; Stocking & Swanson, 1993, Armstrong et al, 1998)
- Applications to diagnostic testing are emerging
  - Attribute-hierarchy model (AHM, Gierl & Leighton)
  - ECD-like applications (Huff; Perlman)
  - Principled assessment designs for inquiry (PADI, Wilson & Mislevy)
  - Task modeling (Luecht, Burke, & Devore, Masters & Luecht, Gierl & Leighton; Luecht & Gierl)

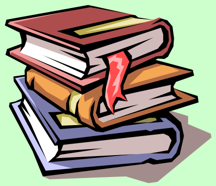
Five AE Processes

- Construct mapping
- Evidence modeling
- Task modeling and construct blueprinting
- Template design and item writing
- Psychometric QC/QA, Calibration, and Scoring

Construct Mapping

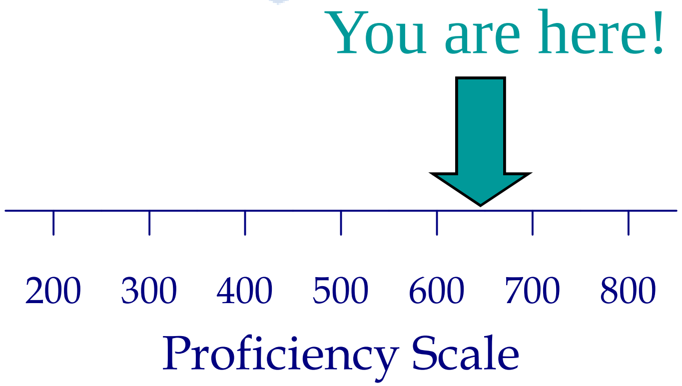
Traditional Views of the Transition from Construct Spaces to Latent Trait Scales





What was the author's purpose?

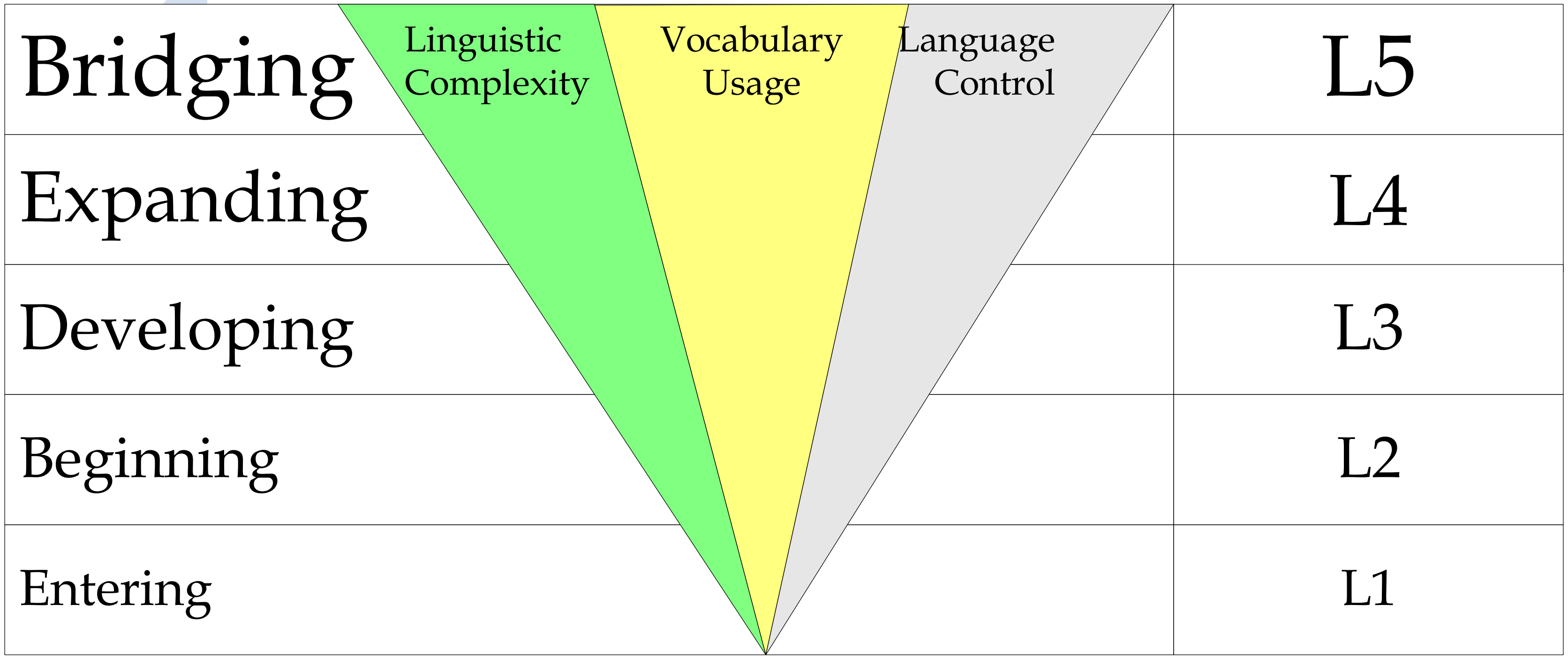
A. To inform
B. To illustrate
C. To persuade
D. To obsfuscate



$$\mathbf{U}_j = \left(u_{1j} = 1, u_{2j} = 1, u_{3j} = 0, \ldots, u_{nj} = 0\right)$$
$$= \left(110\ldots1\right)$$

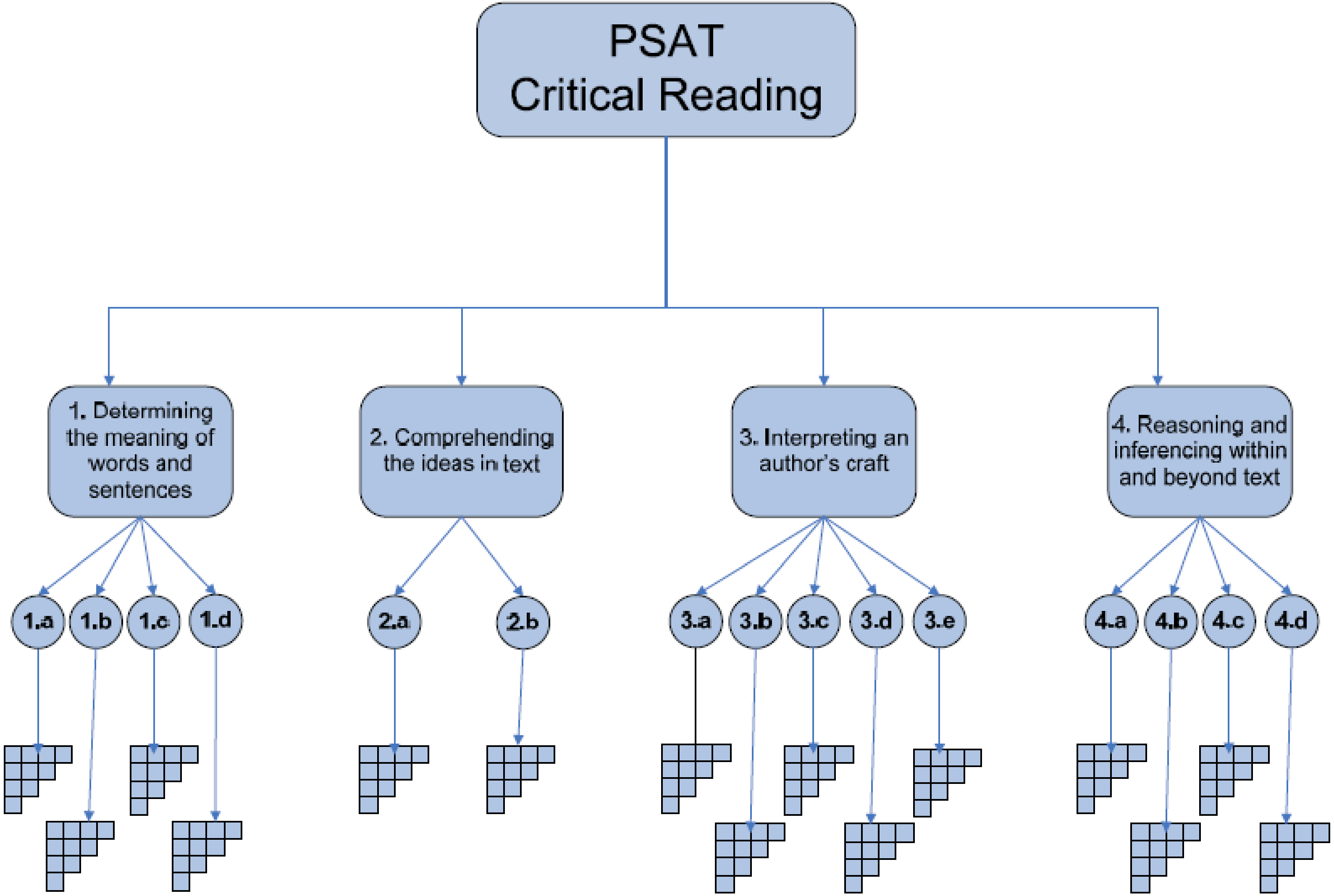
$$\frac{\partial L}{\partial \theta} = \frac{\sum (u - P)}{PQ} \frac{\partial P}{\partial \theta}$$

K-12 Language Proficiency



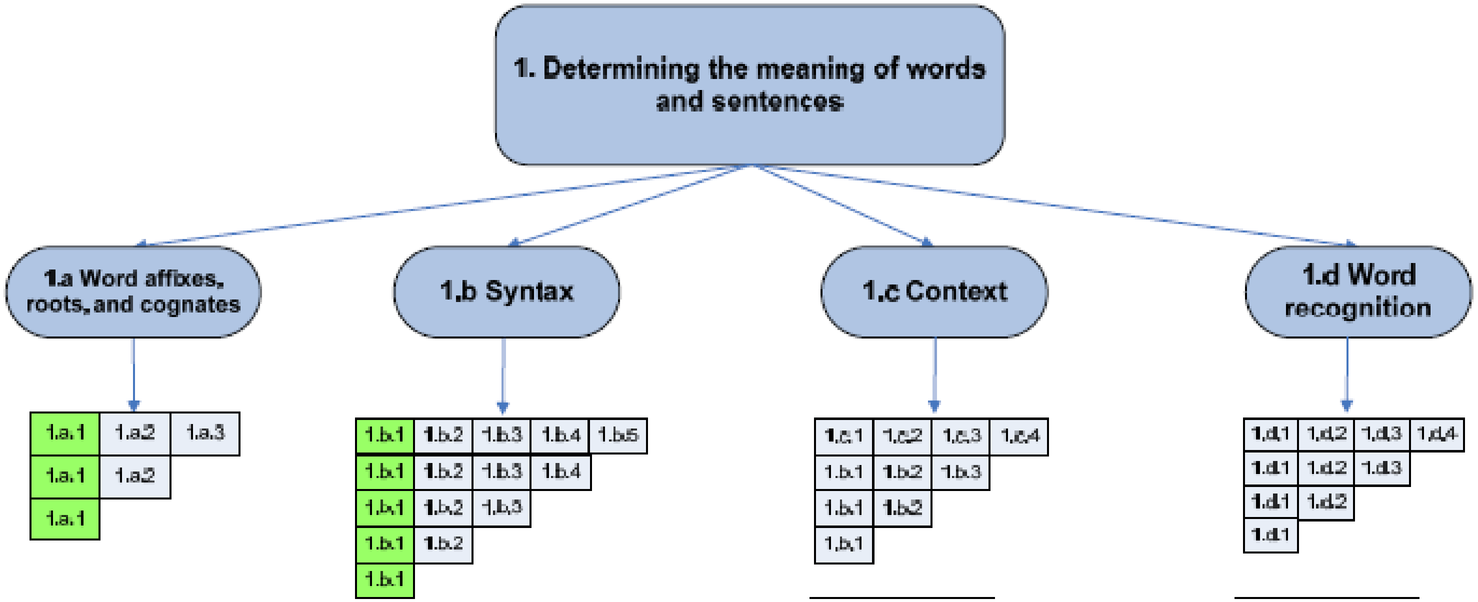
Kenyon, D. (Nov., ,2007). *Examining a large-scale language testing project through the lens of assessment engineering: What can language testers learn?* Keynote address at the Sixth Annual ECOLT Conference, Washington, DC

PSAT Critical Reading



Gierl, M.; Alves, C.; Gotzmann, A.; & Roberts, M. (2008). *Critical Reading PSAT Construct Maps for Cognitive Diagnostic Assessment*. Unpublished Technical Report. Alberta, Canada: Centre for Research in Applied Measurement and Evaluation, University of Alberta

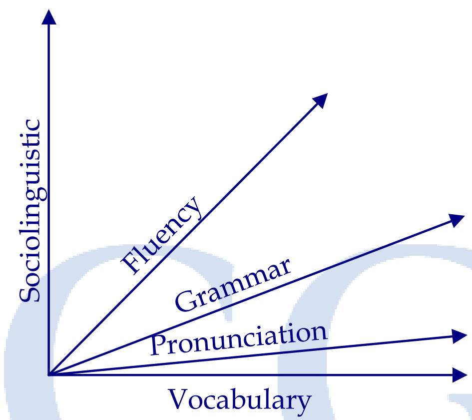
Drilling Down…



Gierl, M.; Alves, C.; Gotzmann, A.; & Roberts, M. (2008). *Critical Reading PSAT Construct Maps for Cognitive Diagnostic Assessmen*t. Unpublished Technical Report. Alberta, Canada: Centre for Research in Applied Measurement and Evaluation, University of Alberta

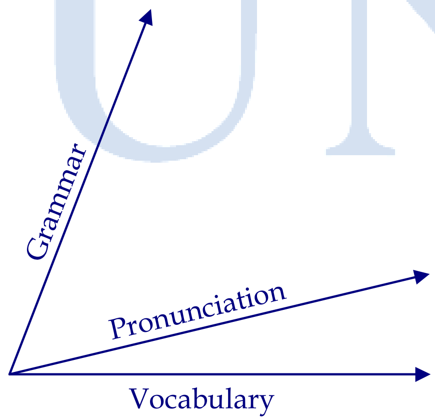
Morphing through Dimensionally More Complex States

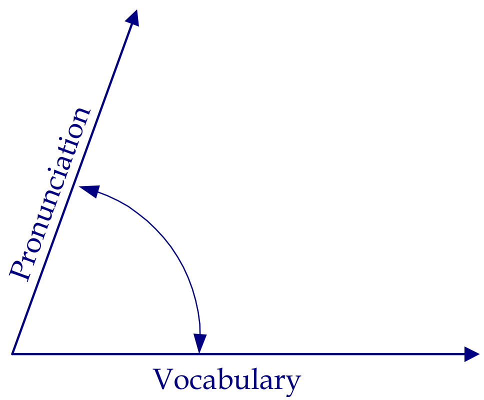
State 3: grammar, pronunciation and vocabulary merge; fluency and sociolinguistics emerge



Due to automaticity, pronunciation and vocabulary become indistinguishable

State 2: Grammar emerges





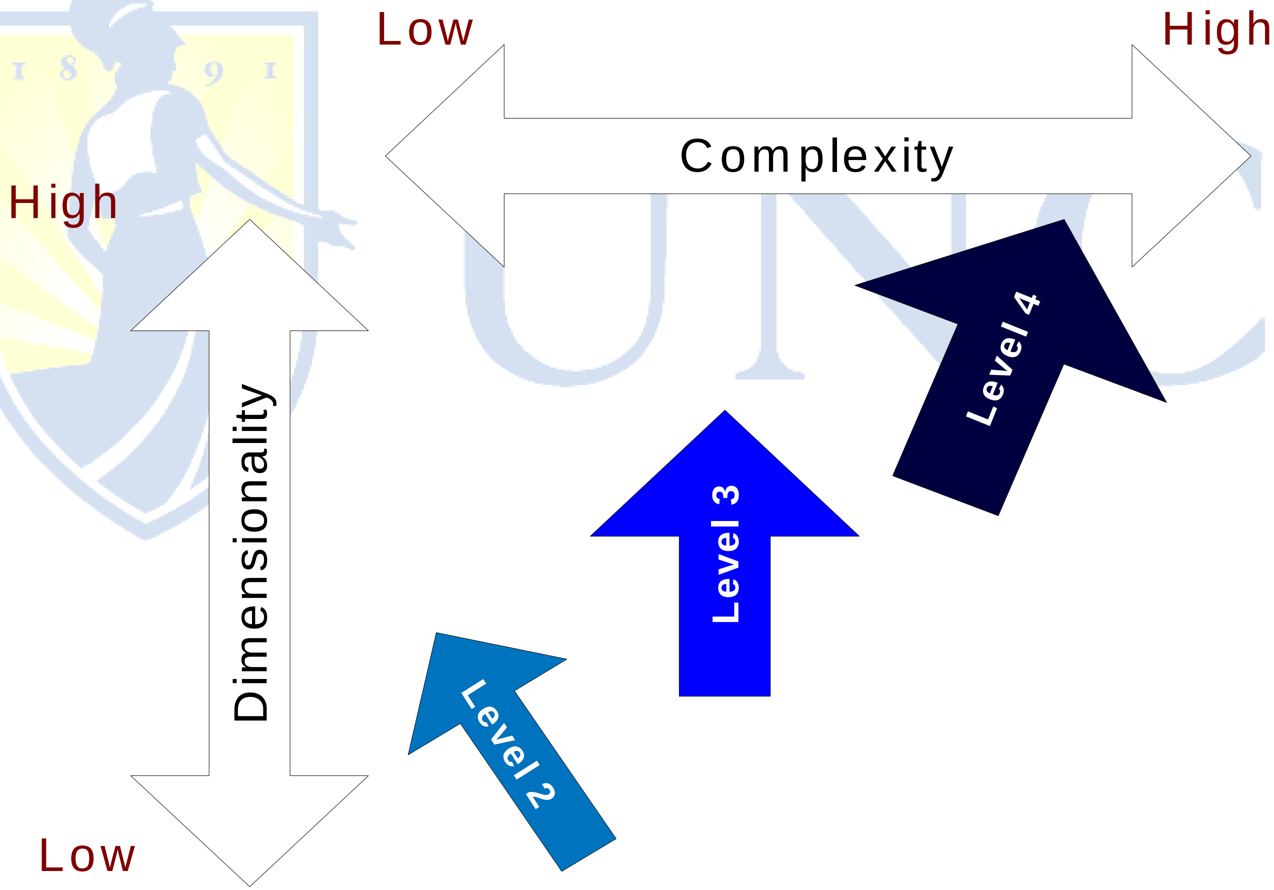
State 1: Pronunciation and vocabulary function autonomously

Luecht, R. M. (2004). Multistage complexity in language proficiency assessment: A framework for aligning theoretical perspectives, test development, and psychometrics. *Foreign Language Annals*, 36(4), 518-526.

Tying Complexity to Cognition

- **Language contexts**
  - **Tasks:** more communication tasks → greater complexity
  - **Topics:** more topics → greater complexity
  - **Information density:** higher structural density of text or speech samples→ greater complexity
- **Cognitive task challenges**
  - **Conceptual Knowledge**: facts, rules, regulations that form the core database for the practitioner
  - **Process Skills**: concrete applications and "how to do [this]"
  - **Evaluation and Synthesis**: reasoning, comparing, contrasting and making inferences or deductions (includes meta-cognition)

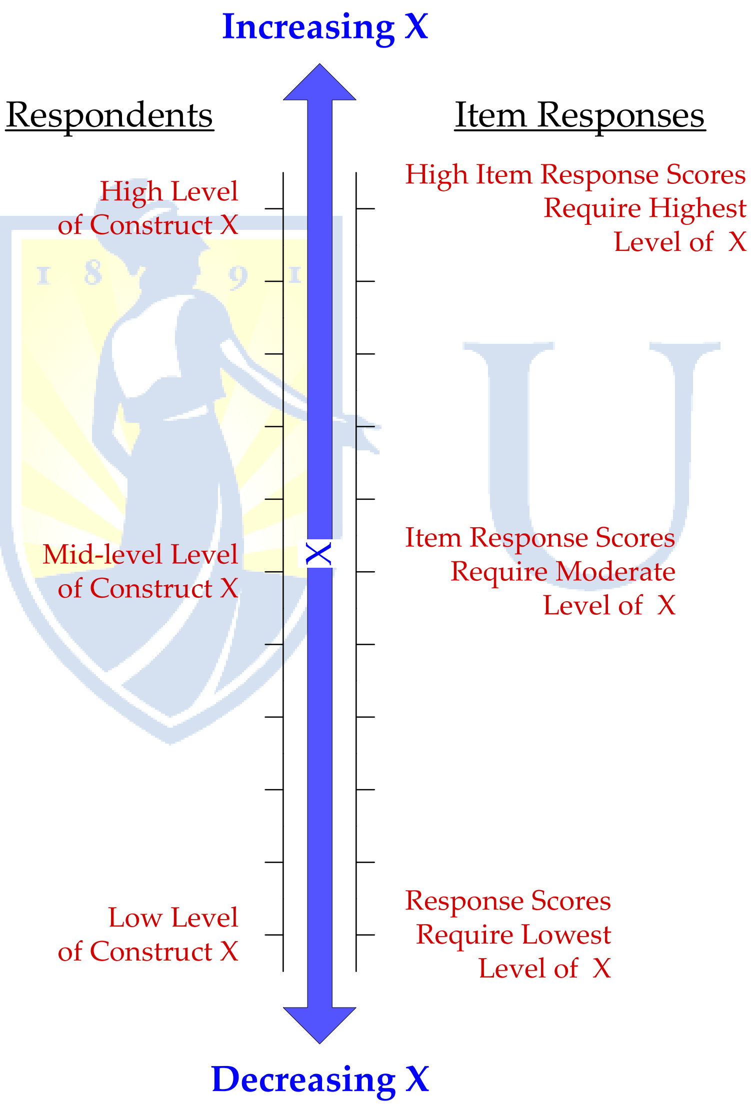
A "New" Perspective on Complexity and Dimensionality



AE and Construct-Based Design

- **Constructs** should be articulated in terms of ordered, hierarchical levels of procedural knowledge and skills, or in terms of levels of cognition applied to well-defined content strands
  - We call the ordered statements that define a construct **claims**" or "**assertions**"
  - Claims are in service of particular decisions along an ordered continuum (fail→pass; 50, 51,…,100, etc.)
- Higher-level claims subsume lower-level claims
- All salient constructs should be specified, along with the expected patterns of relationships among the constructs
- Ultimately…focus on the proficiency claims we wish to make with respect to a specific number of useful, interpretable score scales

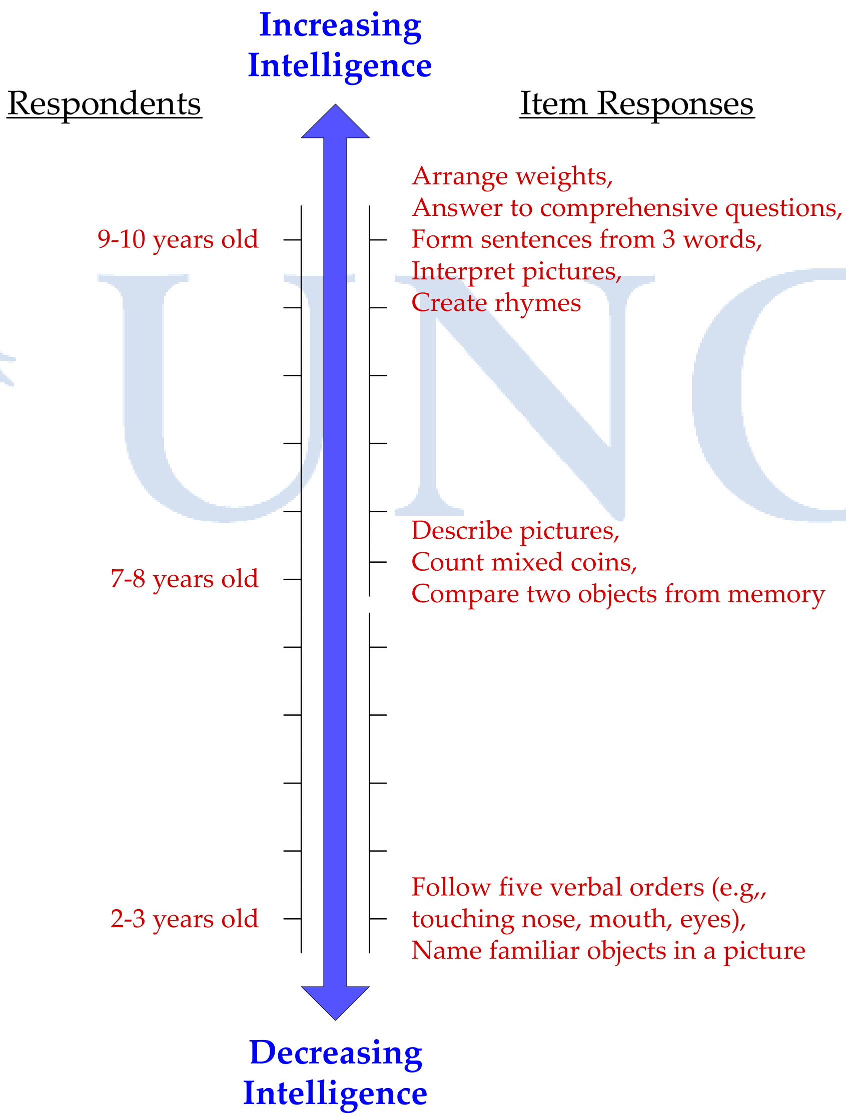
A Construct Map (Wilson, 2005)



"X" can represent a continuum or an ordered set of latent classes

Item locations denote score properties of multiple items with similar characteristics

Binet & Simon (1905): Intelligence



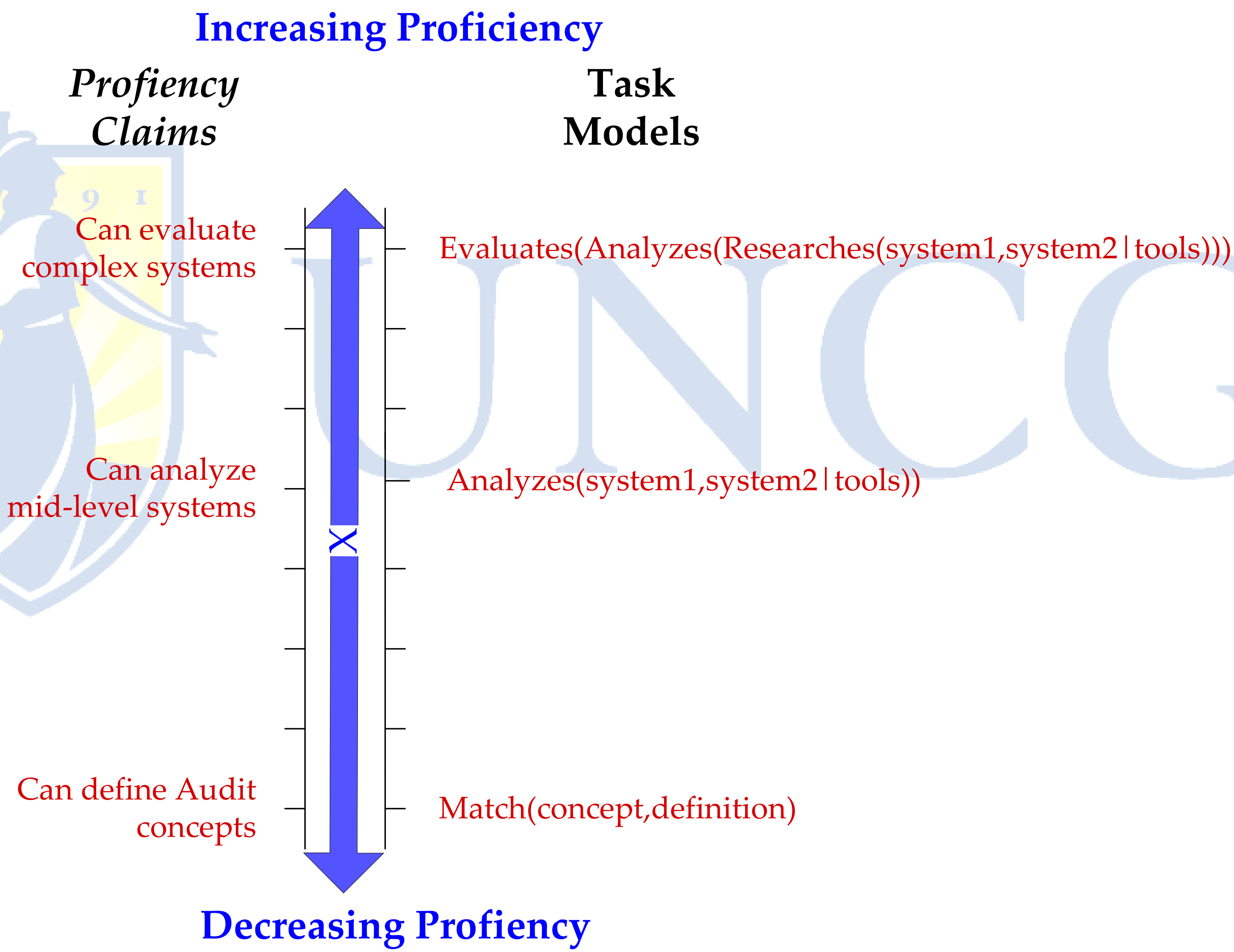
Claims: Examples

- Can *evaluate* the basic distinctions among different <types of entities>
- Can *compare* the effectiveness of components of a system in a specific context
- Can *perform* <appropriate analysis> procedures to assess risk
- Can *prepare* documentation of an operational procedure

What is Construct Mapping (Wilson, 2005; Luecht, 2007)?

- Benjamin Bloom (1956) defined a well-known progression of cognitive skills: knowledge → comprehension → application → analysis → synthesis → evaluation
- Marzano (2000) reformulated the progression as conceptual knowledge (declarative knowledge), process skills (procedural knowledge), and evaluation and synthesis (includes meta-cognition, both declarative and procedural)
- Construct mapping amounts to clearly documenting a progression of <u>ordered</u> claims about proficiencies and skills and the required <u>observable</u> evidence needed to make those claims

Claims and Construct Maps

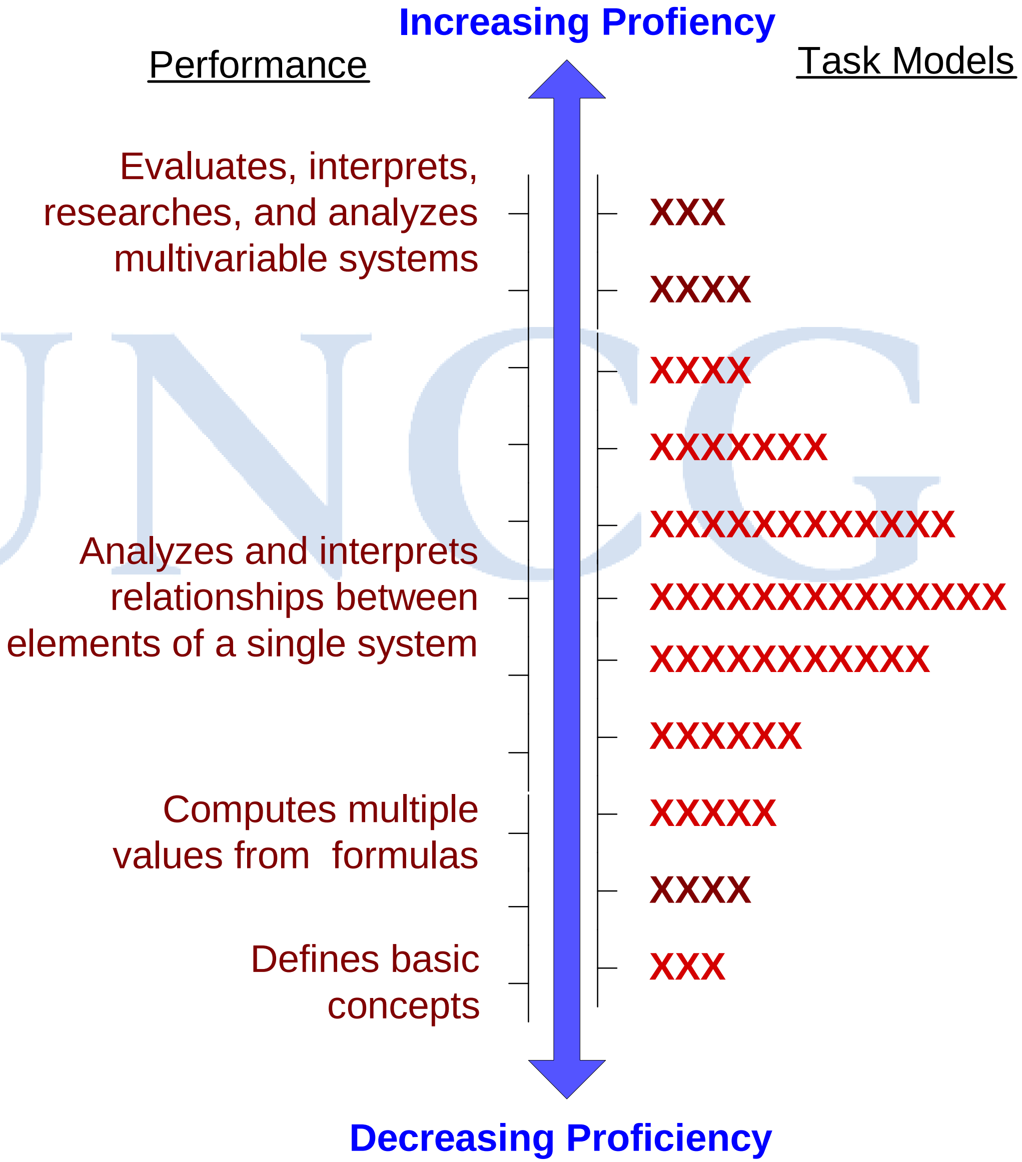


Construct-Based Validation is NOT New (Messick, 1994)

"A construct-centered approach [to assessment design] would begin by asking what complex of knowledge, skills, or other attributes should be assessed, presumably because they are tied to explicit or implicit objectives of instruction or otherwise valued by society. Next, what behaviors or performances should reveal those constructs, and what tasks or situations should elicit those behaviors?" (p. 16)

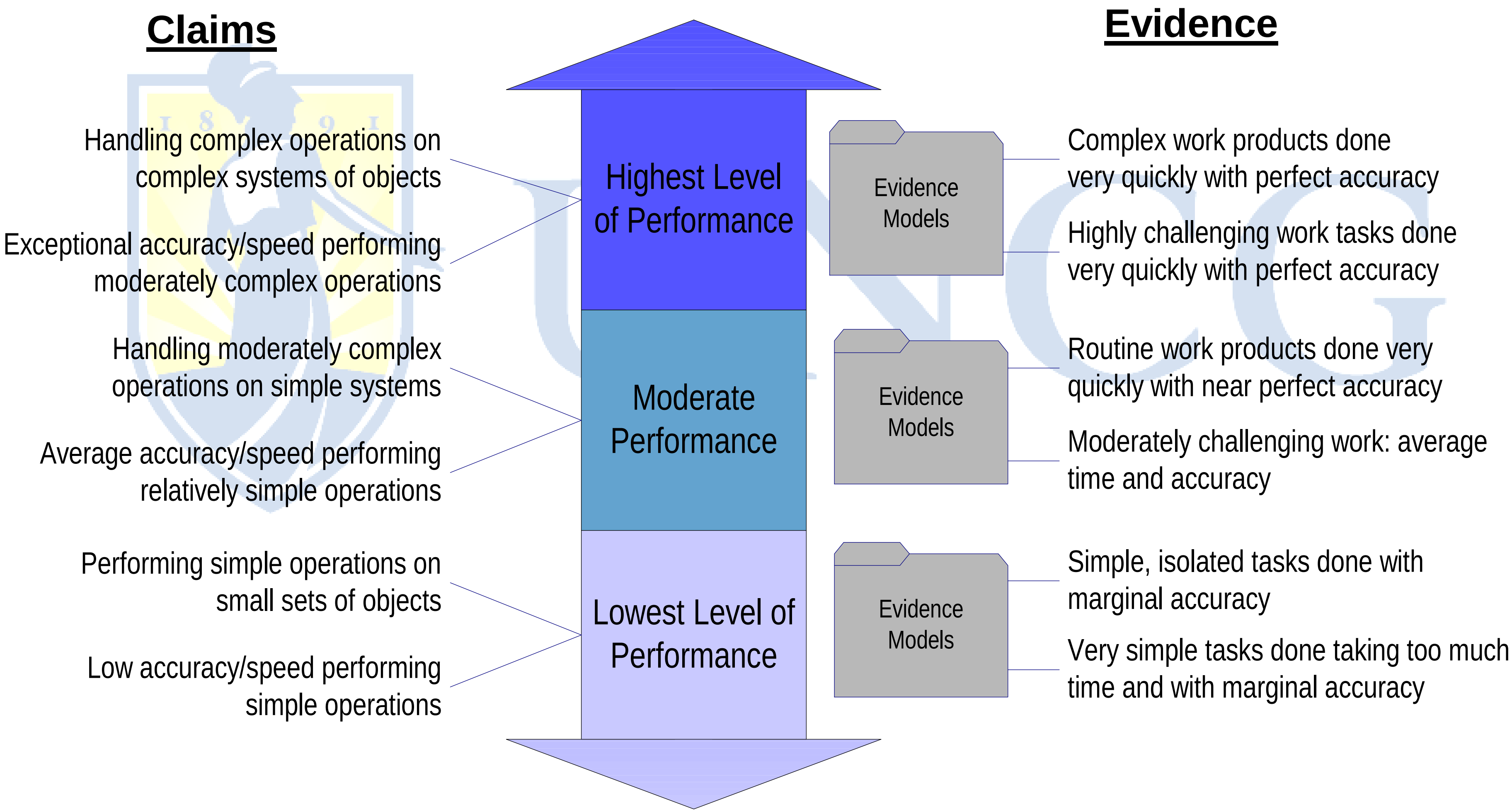
Scores→Scales→Construct Maps

	Assessment Tasks							
Pattern #	1	2	3	4	5	6	7	8
1	4	4	4	4	4	4	4	4
2	3	4	4	4	4	4	4	4
3	3	3	4	4	4	4	4	4
4	3	3	3	4	4	4	4	4
5	3	3	3	3	4	4	4	4
6	3	3	3	3	3	4	4	4
7	3	3	3	3	3	3	4	4
8	3	3	3	3	3	3	3	4
9	3	3	3	3	3	3	3	3
10	2	3	3	3	3	3	3	3
11	2	2	3	3	3	3	3	3
12	2	2	2	3	3	3	3	3
13	2	2	2	2	3	3	3	3
14	2	2	2	2	2	3	3	3
15	2	2	2	2	2	2	3	3
16	2	2	2	2	2	2	2	3
17	2	2	2	2	2	2	2	2
18	1	2	2	2	2	2	2	2
19	1	1	2	2	2	2	2	2
20	1	1	1	2	2	2	2	2
21	1	1	1	1	2	2	2	2
22	1	1	1	1	1	2	2	2
23	1	1	1	1	1	1	2	2
24	1	1	1	1	1	1	1	2
25	1	1	1	1	1	1	1	1



Evidence Models

Construct Maps and Evidence Models



Evidence Models

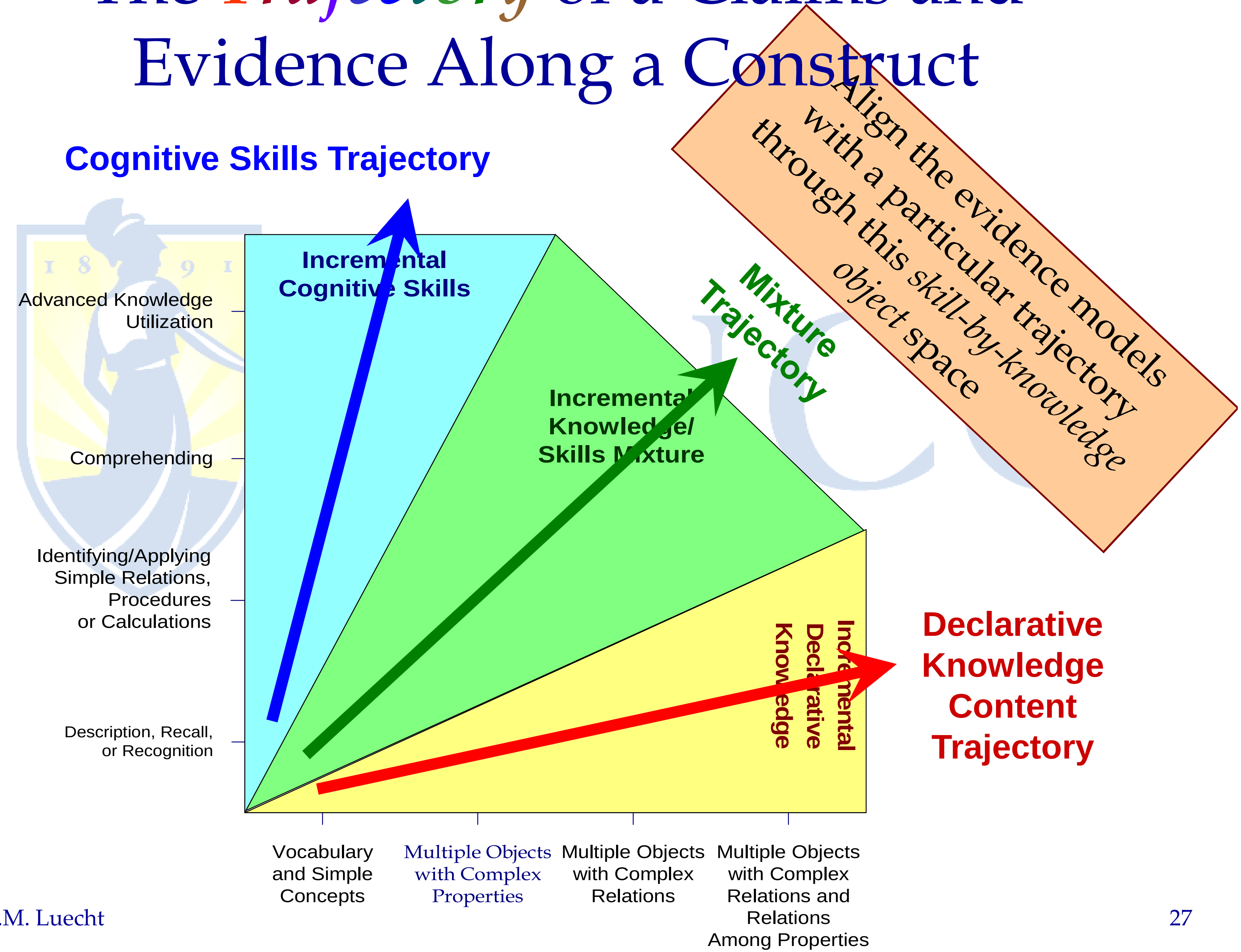
- An evidence model is a documented specification of the ***universe*** of tangible actions, responses, and/or products that would qualify as evidence for a particular proficiency claim…it is a repository of plausible performance tasks for every claim
- Each claim should have one or more evidence models
- Task models are composed directly from the evidence models
- Components of an evidence model include
  - Valid settings or contexts
  - The plausible range of challenges for the target population
  - Relevant actions that could lead to a solution
  - Dangerous or inappropriate actions
  - Legitimate auxiliary resources, aids, tools, etc. that can be used to solve the problem
  - Concrete exemplar products of "successful performance"

Using Practice Analysis Skills (S) and Tasks (T) to Map Evidence Models to a Research & Analysis Construct for Accounting

Skill=	S435	Constr=RAI	LOW	n(Tasks)=	2:	T088	T089								
Skill=	S430	Constr=RAI	LOW	n(Tasks)=	18:	T072	T073	T076	T083	T084	T085	T086	T087	**T088**	
Skill=	S447	Constr=RAI	LOW	n(Tasks)=	2:	T087	T110								
Skill=	S432	Constr=RAI	LOW	n(Tasks)=	20:	T065	T070	T074	T075	T077	T078	T080	T081	**T097**	
Skill=	S431	Constr=RAI	LOW	n(Tasks)=	14:	T065	T072	T082	T083	T084	T087	T089	T097	**T103**	
Skill=	S433	Constr=RAI	LOW	n(Tasks)=	9:	T069	T072	T085	T086	T088	T089	T093	T094	**T109**	
Skill=	S437	Constr=RAI	LOW	n(Tasks)=	6:	T074	T085	T086	T093	T105	T109				
Skill=	S459	Constr=RAI	MED	n(Tasks)=	5:	T089	T091	T094	T095	T122					
Skill=	S439	Constr=RAI	MED	n(Tasks)=	20:	T069	T073	T074	T075	T076	T077	T078	T080	**T082**	
Skill=	S441	Constr=RAI	MED	n(Tasks)=	5:	T071	T072	T085	T105	T106					
Skill=	S445	Constr=RAI	MED	n(Tasks)=	3:	T089	T095	T109							
Skill=	S446	Constr=RAI	MED	n(Tasks)=	8:	T086	T087	T091	T095	T103	T105	T109	T118		
Skill=	S426	Constr=RAI	MED	n(Tasks)=	5:	T070	T076	T085	T093	T106					
Skill=	S414	Constr=RAI	MED	n(Tasks)=	10:	T065	T069	T070	T073	T074	T076	T085	T092	**T103**	
Skill=	S440	Constr=RAI	MED	n(Tasks)=	13:	T071	T072	T080	T089	T091	T093	T094	T095	**T096**	
Skill=	S448	Constr=RAI	MED	n(Tasks)=	19:	T080	T081	T082	T083	T084	T086	T087	T089	T092	
Skill=	S442	Constr=RAI	MED	n(Tasks)=	30:	T065	T069	T070	T073	T074	T075	T077	T078	T080	
Skill=	S438	Constr=RAI	MED	n(Tasks)=	23:	T063	T073	T080	T081	T082	T083	T084	T089	**T091**	
Skill=	S458	Constr=RAI	HIGH	n(Tasks)=	22:	T063	T071	T072	T080	T085	T086	T087	T089	T092	
Skill=	S443	Constr=RAI	HIGH	n(Tasks)=	21:	T082	T083	T084	T086	T089	T092	T093	T094	**T095**	

Luecht , R.M. (March, 2008). *The Application of Assessment Engineering to and Operational Licensure Testing Program.* Invited paper presented at the Annual Meeting of the Association of Test Publishers, Dallas, TX.

The *Trajectory* of a Claims and Evidence Along a Construct



Task Modeling and Construct Blueprinting

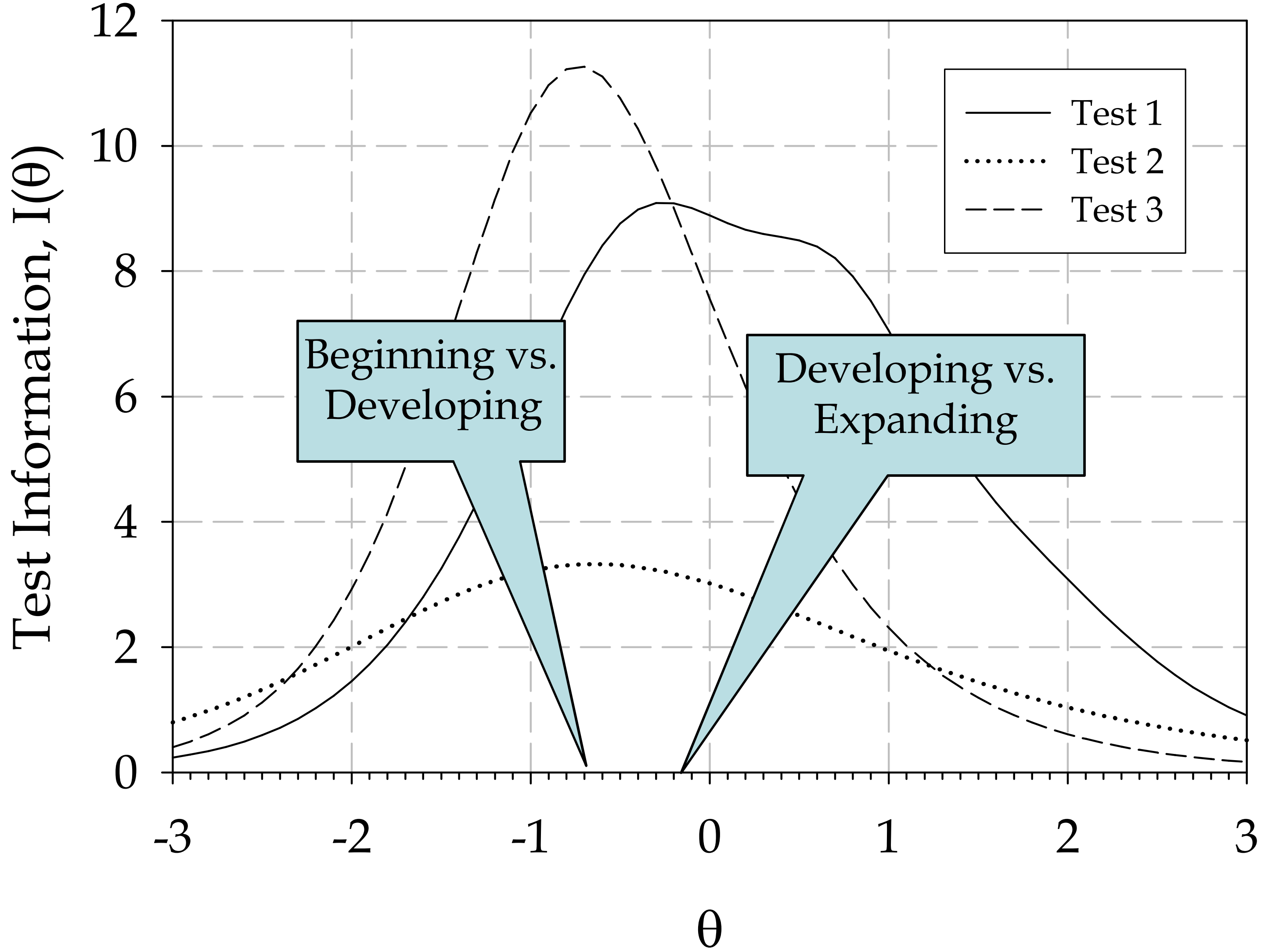
Construct Maps and Targeted Measurement Information

- Measurement information is largely a function of two statistical characteristics of assessment tasks
  - The **difficulty** of each item (i.e., its "location" with respect to some score scale)
  - The **sensitivity** of the item to the underlying construct being measured (i.e., discriminating power of the item)
- We can TARGET measurement information where it is needed most by controlling the difficulty of the assessment tasks
- Under AE, we must jointly control sensitivity to the construct of interest and "nuisance" dimensionality via **task models** and **templates**

Features of Test Measurement Information

- Each item contributes a unique amount of <u>information</u> at specific score values
  - The item information functions are independent of one another for different items
  - A TIF does not depend on any particular items being included in the test
- Under scaling methods such as IRT, the test information functions are directly proportional to the error variance associated with the estimates of θ (EAPs or MLEs)

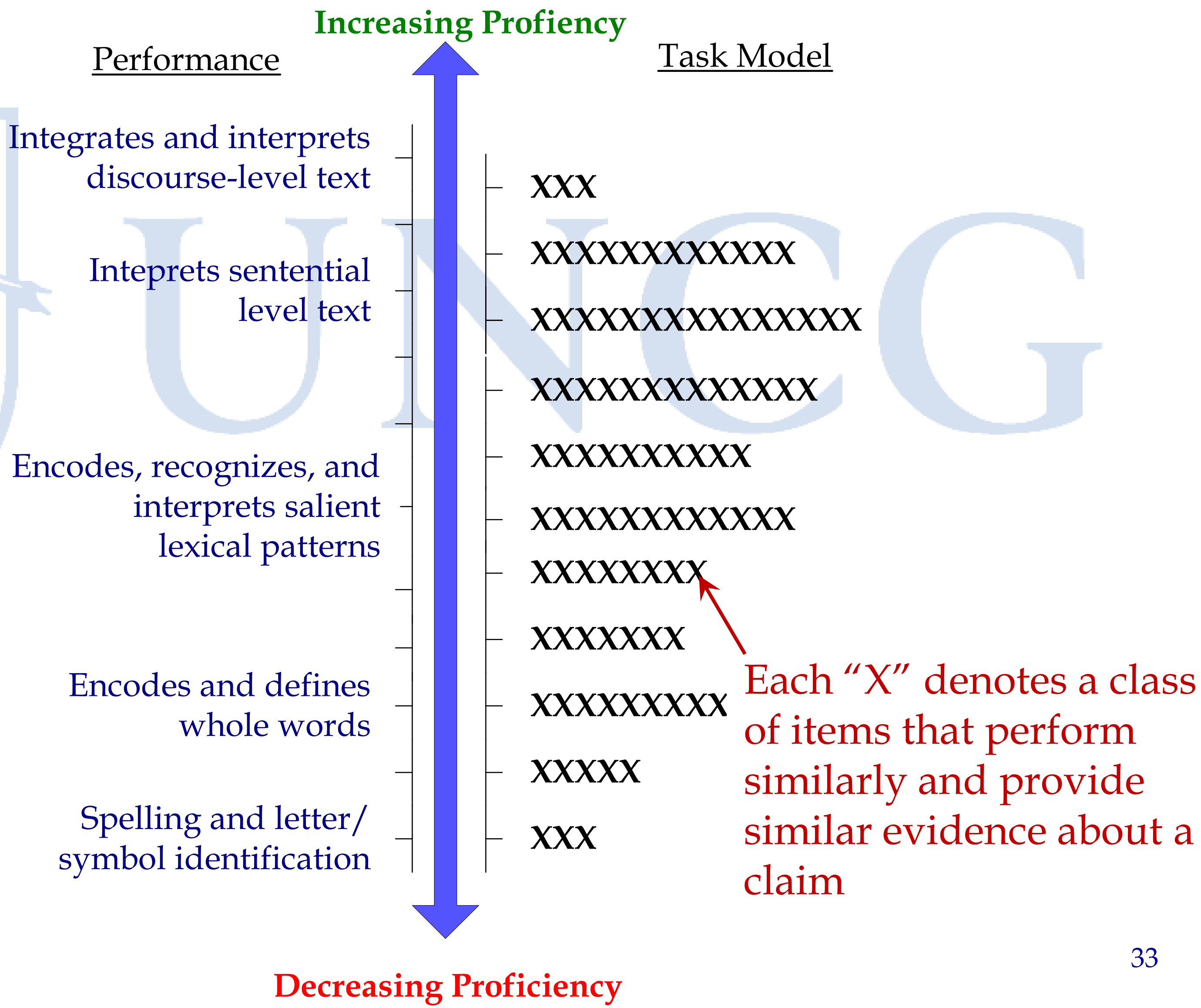
Target Information Functions Tied to ***Decisions***



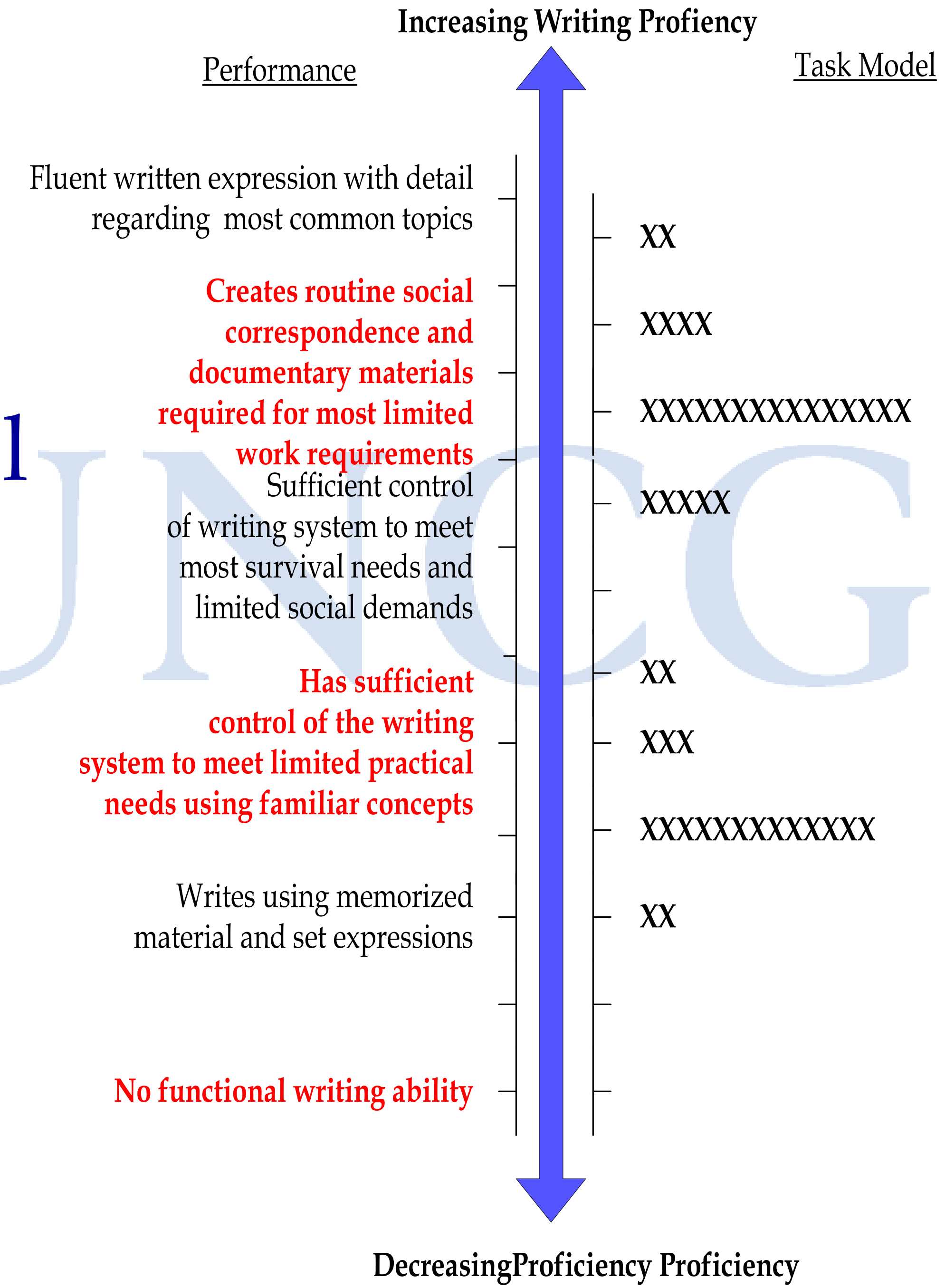
How AE Works to Target Measurement Information

- Measurement precision is targeted to specific regions of the **construct map**
- **Evidence models** define the universe of knowledge and skill tasks that might provide credible, observable, and concrete evidence about the proficiency claims at various levels of the construct
- **Task models** are composed from evidence model components and are stacked in the greatest numbers where to approximate the density of measurement precision needed
- Multiple **task templates** are constructed and empirically validated for each task model
- Task templates are used by item writers to generate exchangeable performance assessment tasks to meet demands

Density of Task Models Proportion to Measurement Precision Needs



A Construct Blueprint with a Task Model Distribution of Measurement Opportunities for Writing Proficiency



Task Models: A New Way to Blueprint

- Task models describe THREE characteristics in terms of conjunctive performance statements stated on a particular construct map
  - Objects and their properties
  - Nature of relationships among objects and their persistence (e.g., hierarchical, directional, causal)
  - Functional clauses represent the action required on the objects and any specified conditions; e.g., ***Action*(Object1, Object2)** or ***Action*(Object|conditions)**
- Cognitively complex tasks can be represented by higher-order functional clauses (e.g., "*Maintains*()" versus "*Updates*" or as nested primitive functional clauses)
- A useful task model should be capable of producing multiple templates; however, all of the templates for a given task model should be empirically shown to behave similarly in terms of their psychometric properties

Task models aligned on the construct map replace traditional content blueprints. For example, NO MORE….

Content Areas	*Knowledge & Concepts*	*Applications*	*Evaluation & Synthesis*
A	*8%*	*10%*	*12%*
B	*6%*	*6%*	*8%*
C	*8%*	*10%*	*12%*
D	*6%*	*6%*	*8%*

Defining and Validating Task Models

- Task models differ in ***location*** (difficulty) along the construct map
- Each model provides ***measurement information*** in a particular region of the construct map
- Deficits or gaps are filled by adding <u>more</u> task models
- Ordering of task models must be ***empirically*** confirmed

Cognitive Elements of Task Models

- Declarative knowledge manipulatives
  - Vocabulary/popularity of words
  - Number of objects (numeric entities, actors, concepts, or idea units) and extent of details
  - Relationships among objects
  - Relationships among properties of objects
- Procedural-skill manipulatives
  - Describing using objects simple recognition and recall
  - Interpretation, translation, calculations, procedures-by-rote, or identifying simple systems of relations
  - Comprehension: relating knowledge structures and predicting outcomes
  - Advanced utilization of knowledge (synthesis, evaluation, and advanced applications using complex knowledge structures)

Building Task Models that Control Difficulty and Dimensionality

- Controlling the number of key objects
- Identifying key properties of the objects relevant to the task (facilitative or distractive)
- Controlling the number of objects to be acted upon or manipulated
- Constraining the number and nature of the relationships
- Specifying and controlling the cognitive level of the action(s) or manipulation(s) required by the task
- Explicitly defining the nature and nesting of relations among objects
- Explicitly defining the nature and hierarchical sequencing of functional clauses

Sample Task Model Worksheet

Task Model Specification Worksheet

Construct Identifier:	Applied statistics and educational measurement statistics	
Level(s) of Construct:	Basic	
Primary Context:	Effect size, *d*	
Competency Claim:	Computes and interprets an effect size as a standardized difference between groups or levels of an independent variable	
Evidence Documentation		
1.	Successfully computes *d*, given two means and std. deviations from a common population	
2.	Successfully computes *d*, given two means and std. deviations from independent populations (i.e., using the pooled variances)	
3.	Correctly interprets *d*, given two means and std. deviations from a common population	
4.	Correctly interprets *d*, given two means and std. deviations from independent populations (i.e., using the pooled variances)	
Conceptual Task Models		
	Specific Tasks	*Expected Mastery Criteria*
1.	***interprets***(*d*|single pop. means)	Plausible choice from options
2.	***interprets***(*d*|separate pop. means)	Plausible choice from options
	interprets(*d*|levels of indep. variable)	Plausible choice from options
3.	***computes***($d\|\mu_1, \mu_2, \sigma$)	Correct value
4.	***computes***($d\|\mu_1, \mu_2, \sigma_1, \sigma_2$)	Correct value
5.	***interprets***(***computes***($d\|\mu_1, \mu_2, \sigma$))	Plausible choice from options
6.	***interprets***(***computes***($d\|\mu_1, \mu_2, \sigma_1, \sigma_2$))	Plausible choice from options
7.	***interprets***(**generates**(*scatterplot*|*x*,*y*))	Plausible choice from options
	Manipulable Features of Complexity/Difficulty	
1.	Magnitude of *d* (low, moderate, high)	
2.	Standardization of variables	
3.	Number of groups (two or more)	
4.	Sign of the effect size	
5.	Formulas provided	
6.	Software/calculator access/training	
7.	Graphic facilitators (depictions of the distributions)	
	Features Irrelevant to Complexity/Difficulty	
1.	Variable labels	
2.	Magnitude of scale	
3.	Compute vs. interpret vs. interpret(compute())	

Rules for Building Task Models

- Task models should be **incremental**–that is, ordered by ***complexity***
  - Numbers knowledge objects
  - Depth of salient knowledge object properties
  - Extent of salient relations among objects
  - Sequential or simultaneous actions required to successfully complete the task
- Task models are the same level must reflect be **conjunctive performance**
- Higher performance assumes that lower level knowledge and skills have been successfully mastered

Task-Model Grammar (TMG) (Luecht in progess)

- **Knowledge objects** and their properties describe key task entities
  - Format: ***Object*.property.value="data"**
  - Drivers
    - ✓ Number of objects
    - ✓ Number of manipulated properties
    - ✓ Popularity/familiarity of the objects
- ***Relational operations*** link two or more objects
  - Format: ***IsRelated*(Object1, Object2, Nature_of_relationship)**
  - Drivers
    - ✓ Number of objects related
    - ✓ Nature of the relationship
    - ✓ Nesting of relations
- ***Functional clauses*** express an action or operation
  - Format: ***Action*(Object1, Object2)** or ***Action*(Object|conditions)**
  - Drivers
    - ✓ Number of arguments
    - ✓ Complexity of the function
    - ✓ Nesting of functions

Language-Based Task Design Drivers to Consider Under TMG

Knowledge

- Unique vocabulary/TTR
- Discipline-specific vocabulary
- Grammatical structures
- Semantic relations
- Number of "idea units"
- Key properties of objects
- Nature of relations
- Graphic complexity
- Contextual constraints/setting details
- Formula familiarity
- Auxiliary language

Cognitive Skills

- Auxiliary aids
- Training/direction
- Calculation complexity
- Persistence of relations
- Mental manipulations of images and visual objects
- Derivation or manipulation of formulas
- Functional constraints on applications (e.g., open-ended functionality vs. tight scripting)

Calibrating Task Models

- The task model is treated as a family of items with similar operating characteristics
- A hierarchical Bayesian framework can be used to estimate the task model parameters
  - Hyperparameters are employed
  - Uncertainty is automatically factored in
- Scoring uses the joint probability distribution
  - Less statistically efficient than separate item parameters
  - More efficient in terms of operational scoring

From Task Models to Templates

- Each task model should yield **multiple templates**
- Templates are elaborated "item models" used to ***render*** and ***score*** the items in a "family"
  - Each template has a formal data structures that captures the fixed and variable features of the task model
  - Each template "scoring evaluators" that specify how measurement opportunities are converted to "scores" such as 1=correct, 0=incorrect
- Templates must be empirically validated to ensure that they are controlling difficulty and extraneous sources of noise

Template Design and Item Writing

AE-Based Templates

- Each **task model** can be represented by multiple, exchangeable **templates**
- A **template** has three components
  - Rendering model: detailed presentation format data and constrained interactive components for each task (e.g., LaDuca, 1994; Case & Swanson, 1998; Luecht, 2001, 2006)
  - Scoring evaluator: produces item- or measurement-opportunity-level scores from a performance (Luecht, 2001, 2005, 2006)
  - Data model: represents the rendering model, scoring evaluator, associated difficulty drivers (radicals), and incidental surface-level manipulables in database structures that can be used/activated by item writers to generate two or more items

Item Model (LaDuca, 1994)

A 19-year old archeology student comes to the student health service complaining of severe diarrhea, with large-volume watery stools per day for 2-days. She has no vomiting, hematochezia, chills, or fever, be she is very weak and very thirsty. She just returned from a 2-week trip to a remote Central American archealogical research site. Physical examination shows a temperature of 37.2 degrees Centigrade (99.0 F), pulse 120/min., respirations 12/min., and blood pressure 90/50 mm Hg. Her lips are dry and skin turgor is poor. What is the most likely cause of her diarrhea?

A. Anxiety and stress from traveling
B. Inflammatory disease of the bowel
C. An osmotic diarrheal process
D. A secretory diarrheal process
E. Poor eating habits during her trip

from Haladyna, T. (2004). *Developing and Validating Multiple-Choice Test Items*, LEA, p 162.

A Rendering Template

<Patient.article><Patient.description.age>
<Patient.description.occupation>
"comes to" <Setting.description> *"complaining of"*
<Patient.ailment.symptom1>
<Patient.ailment.symptom1.duration>
<Patient.ailment.symptom2>
<Patient.ailment.symptom2.duration>
<Patient.history.activity.recent>
<Patient.physicalexam.temp=# C, (convert(C,F))>
<Patient.physicalexam.pulse=#/min>
<Patient.physicalexam.respiration=#/min>
<Patient.physicalexam.bp=#1/#2>
<Patient.physicalexam.symptom1>
<Patient.physicalexam.symptom2> ***"What is the most likely cause of*** <Patient.ailment.prime_symptom> "?"

A Rendering Template for Simple Statistics

A <*setting.container*> holds <*object1.count=x*> <*object1.description*> <*object2.count=y*> <*object2.description*>, and <*object3.count=z*> <*object3.description*>. If we select <*task.action.select.object_count=k*> <*task.action.select.objectdescription*> from <*setting.container*> , what is <*task.response_object*> that the <*task.action.select.objectdescription*> is <*object1.description*>?

<task.answer.distractor1=1/n, $n=x+y+z$>
<task.answer.distractor2=1/{x,y, or z} >
<task.answer.distractor3={x,y, or z} /{$(x+y)$,$(x+z)$,$(y+z)$}>
<task.answer.correct={x,y, or z} /$(x+y+z)$>

Scoring Evaluators

- A **scoring evaluator** is a software or human "agent" that coverts an examinee's response(s) into a numerical score; this is conceptually similar to Wilson's (2005) "outcome spaces"
- Single-key scoring evaluators typically resolve to a dichotomous or binary score
  - $y_{ij}=f(r_{ij},a_i)$ for single responses
  - $y_{ij}=f(\mathbf{r}_{ij},\mathbf{a}_i)$ for vectors of response variables
  - $y_{ij}\in \{0,1\}$
- Correct answer key (CAK) scoring evaluators use the "correct" answer key(s)
- Incorrect key evaluators are useful for diagnostic scoring (Luecht, 2005, 2006)
- AI-based evaluators are possible (e.g., automated essay scoring)

Data Models

- A data model is a structured representation of the salient rendering template task components and related information needed to compose, administer, and score items that are generated from a particular template
- Plausible values to plug into the rendering template using look-up values or ranges of values
- Constraints on use of tools or auxiliary resources (e.g., calculators, measuring devices) are specified
- Parameters are specified for factors that directly or indirectly affect task difficulty (e.g., extent of intermediate calculations required, information density limits, etc.)
- Content, contexts, and other coded data in the task model are specified
- Values, rubrics, or scripts used by the scoring evaluator become part of the data model
- Automated, adaptive item construction using scripts or callable agents/routines is theoretically possible

Possible Additional Fields in the Data Model for Capturing Task Difficulty and Complexity

- Complexity and difficulty fields based entirely on empirical statistics (e.g., *p*-values, IRT statistics, dimenisonality weights, etc.)
- Complexity and difficulty fields based on item writer/test designer judgments (e.g., taxonomic "cognitive" codes)
- Template data features linked to complexity and difficulty are based on derived, replicable, cognitively relevant task models
  - A representational grammar is used to capture the salient model features
  - Data models are developed and empirically link the features to difficulty and complexity indicators

Empirically Validate the Stored Components for Each Template and Associated Task Model

- Try out prototypes to detect which components affect changes in *difficulty*
- Use statistical quality control (QC) analysis to identify potential sources of "error" and implement template-level controls to reduce such covariance
- Templates should account for a large proportion of explained item difficulty variance

Templates and Item Writing

- All item writing is funneled through one or more templates (i.e., item writers do NOT create their own templates)
- Component palettes can be restricted for each template
- Subtle **variations** in ***templates***, ***component palettes***, and ***content/context*** → lots of possible templates, and by extension, even more items, all with ***similar*** psychometric characteristics

Other Engineering Steps

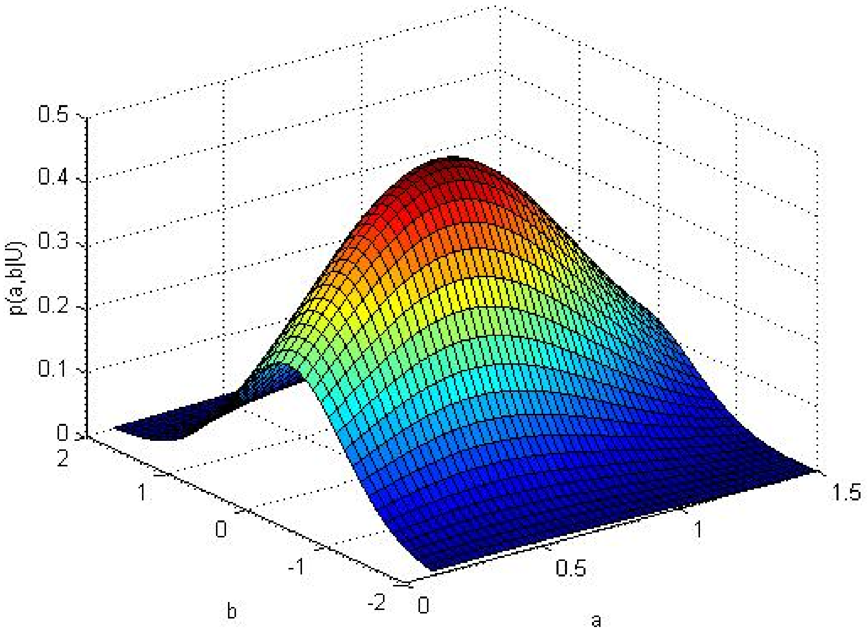
- Create **pricing sheets** to evaluate costs of new templates and component palettes
- Use **cost-benefit analysis** to evaluate
  - The information-per-unit-of-time for costly components
  - Real costs ($$$$) per unit of information
- **Maximize** the number of measurement opportunities and **minimize** the costs
- Make automated test assembly easier
  - Task models match ALL specification (demands)
  - Plenty of simulations (supply) can be generated

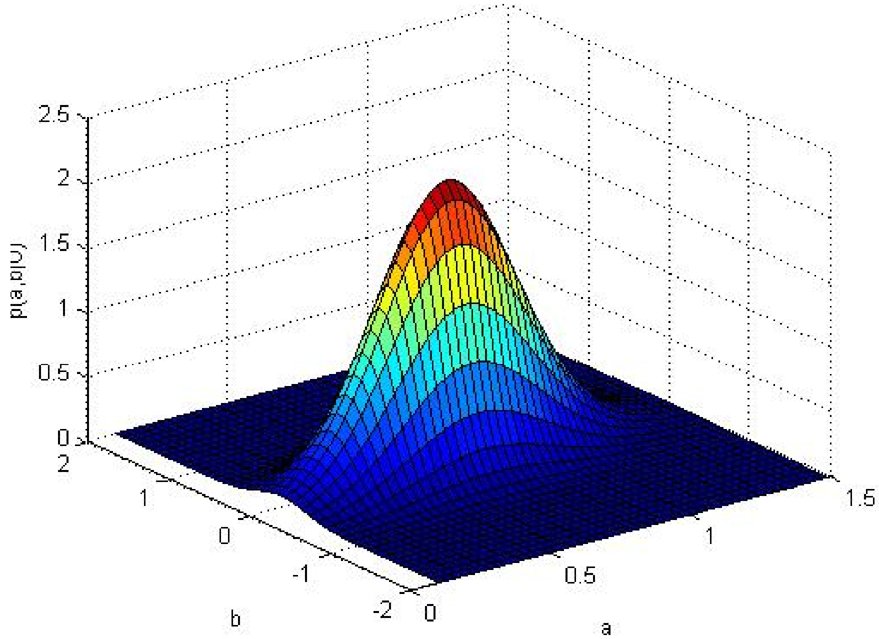
Psychometric QC/QA, Calibration and Scoring

Supporting Psychometrics

- Task models and/or templates can be calibrated instead of individual items, using a hierarchical Bayes framework (Glas & van der Linden, *APM*, 2003)
- Treat the hyperparameters as "super parameters" for the *task model*
- Estimate one set of common means and variance-covariances for the entire family
  - Less pretesting needed, once templates are verified
  - Fewer parameters leads to robust estimation
  - Misfit can be minimized if families are "well formed"
  - Hierarchical framework is extensible as a **QC mechanism**
    - ✓ Minimize posterior variance associated with individual items within templates
    - ✓ Minimize posterior variance associated with templates with task models

QC via the Posterior Distributions for Task Models = P(*a*,*b* | U)

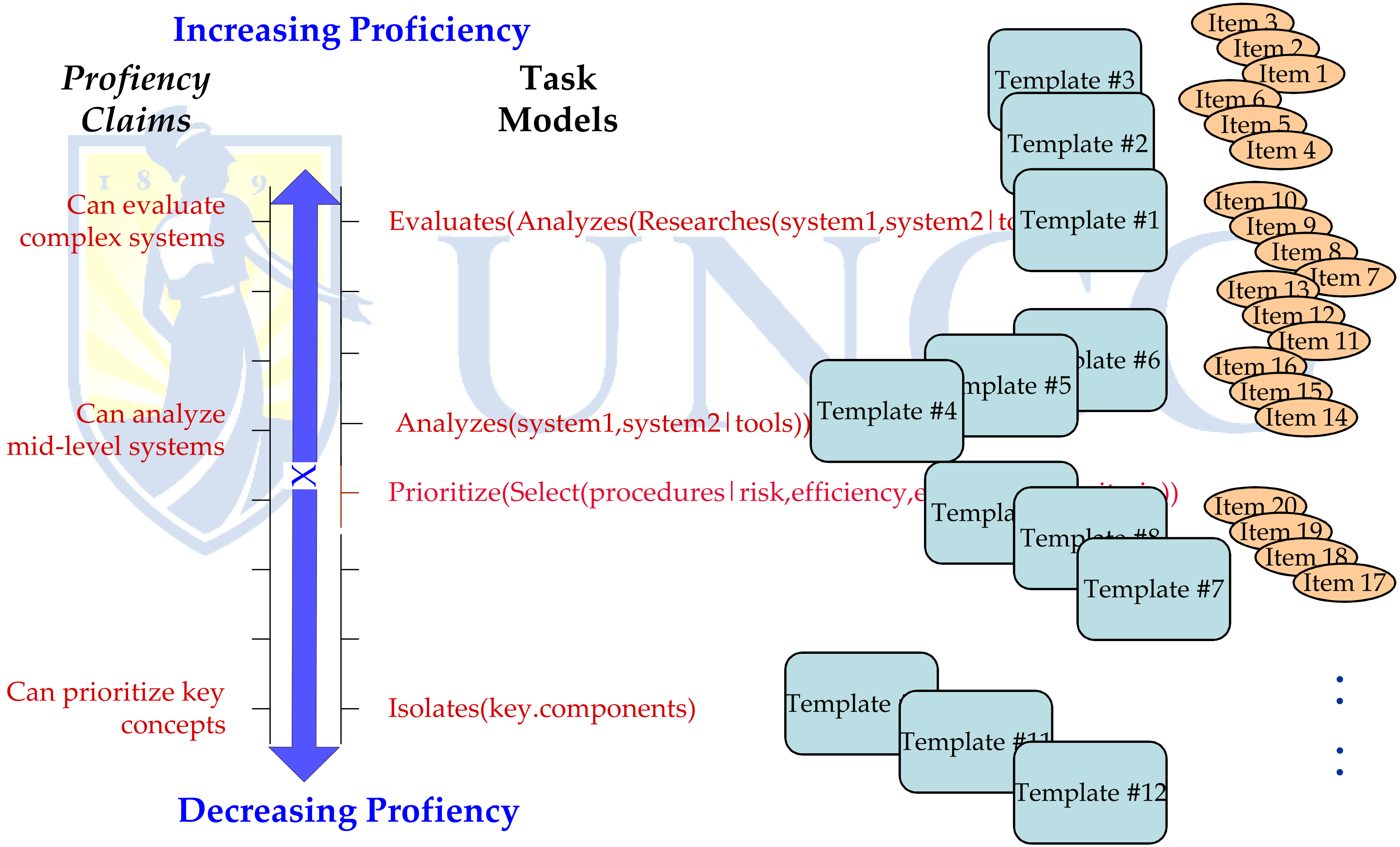




Lower Quality Task Model or Template

Higher Quality Task Model or Template

From Construct Maps to Items

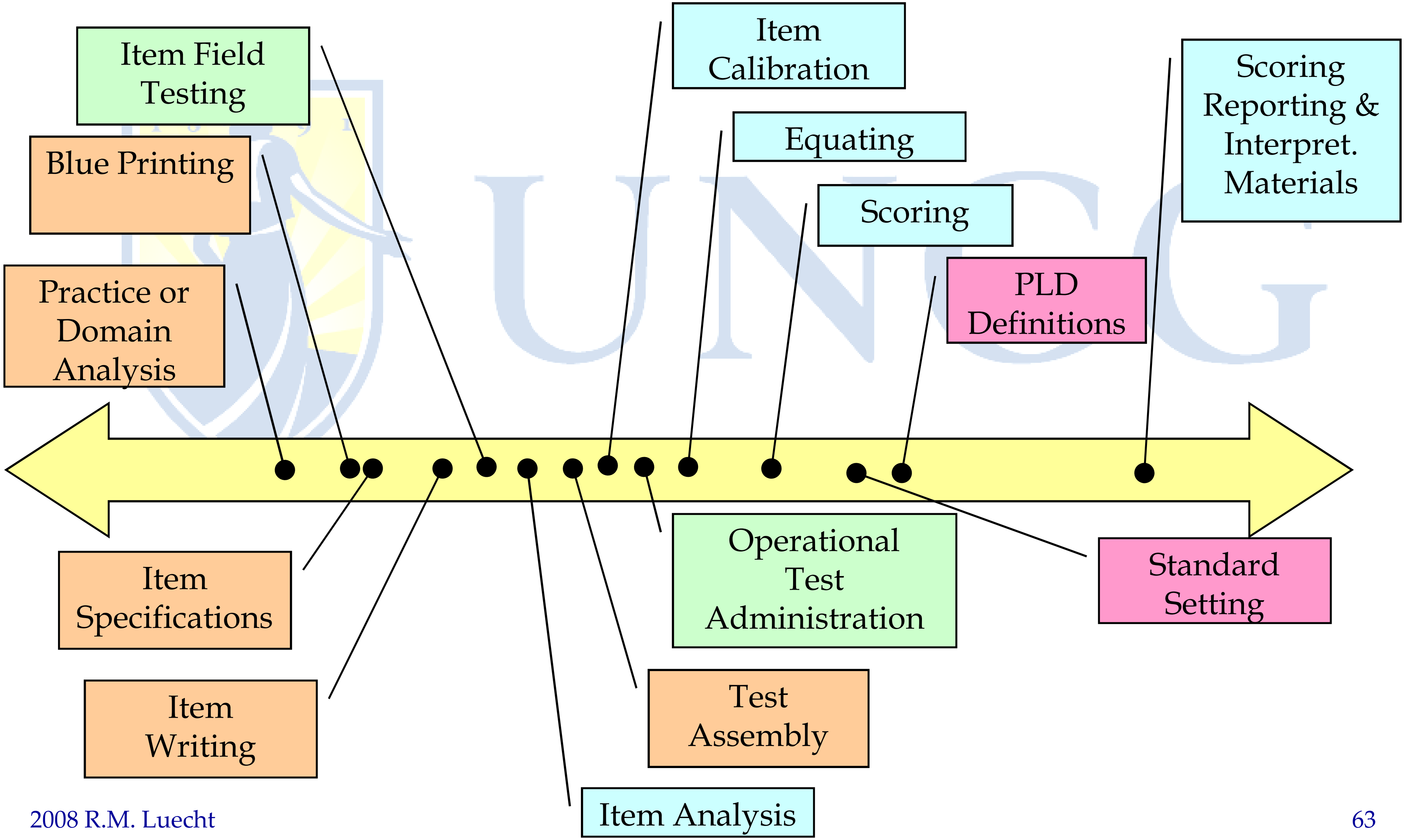


Scoring Paradigms

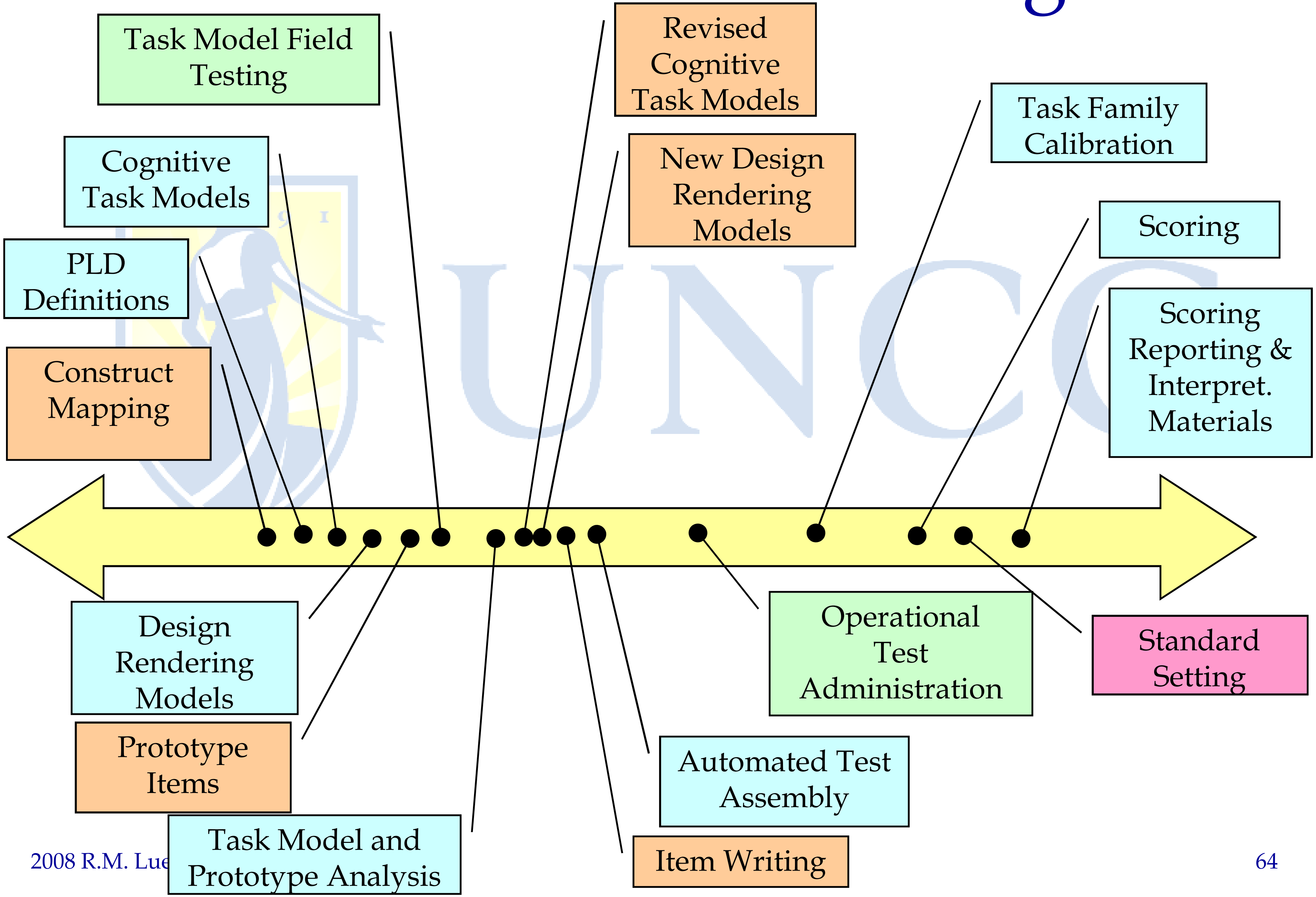
- Hierarchical Bayes
  - Calibrated item statistics can exist at the item, template, or task-model level
  - Integrate over the joint distribution of parameters (see Glas & van der Linden, 2003)
- Multidimensional scoring
  - Separate ability metrics can be maintained
  - Augmented scoring can "steal" collateral information (e.g., *Test Scoring*, Wainer et al, 2001, Ch. 9) but induces a regression bias
  - Full-information MIRT scoring avoids the bias (Segall, 1996, 2000, Luecht, 1996, van der Linden, in progress, Luecht, Gierl, and Ackerman, in progress)
  - Cognitive diagnostic (constrained latent-class) models (e.g., Henson and Templin, 2006, 2007, 2008)

Integrated AE Processes

Traditional View of the Assessment Process



From AE to Std. Setting



Thank you!

rmluecht@uncg.edu
Curry 209, UNCG
PO Box 26170
Greensboro, NC 27402-6170