Think Small (Know the details)

Principle 4 (Operations)

v1.4

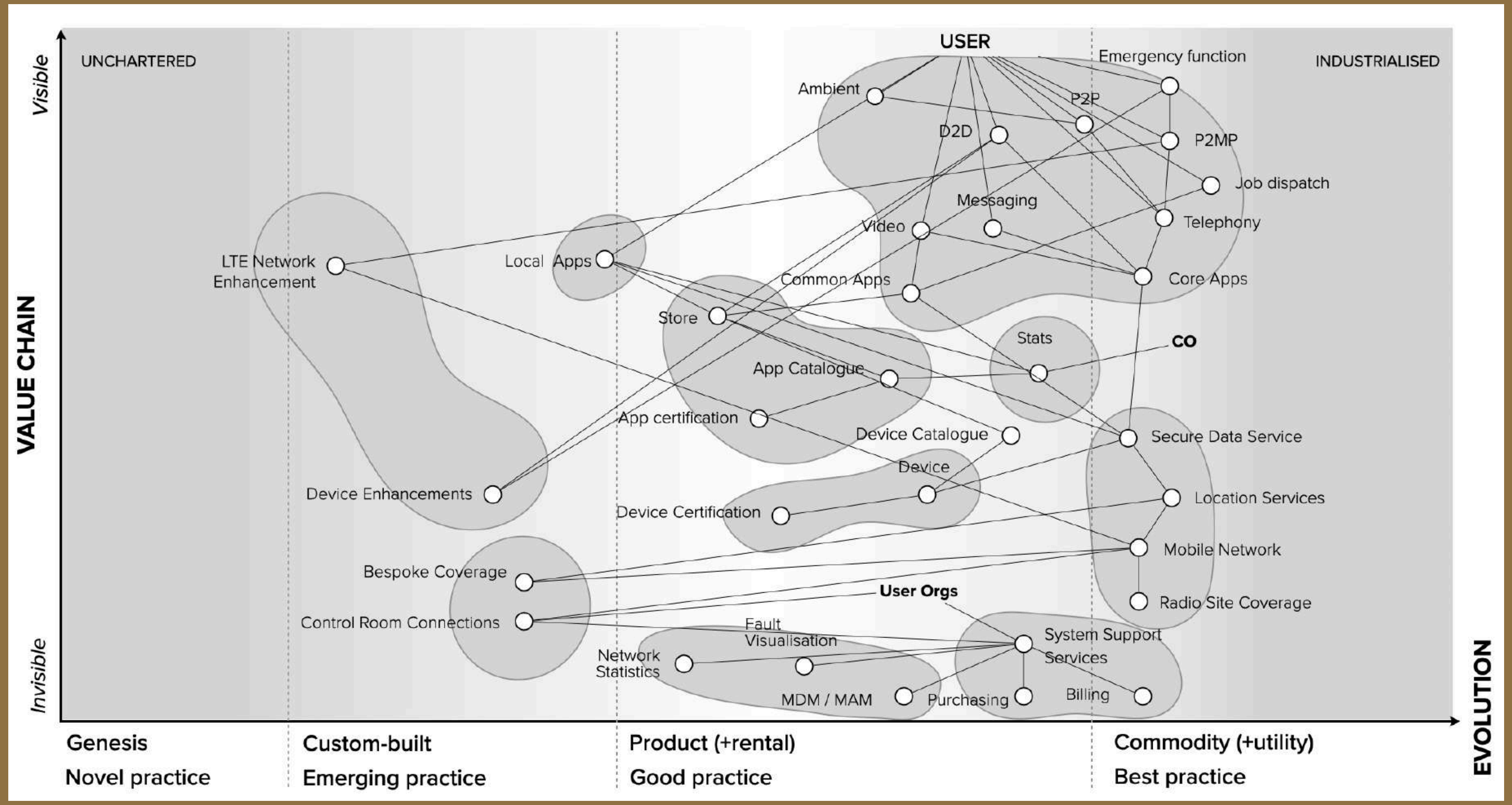
PowerMaps

PRINCIPLES (universally useful ways of operating any organisation can adopt)

CATEGORY	COMMUNICATION	DEVELOPMENT	OPERATIONS	LEARNING	LEADING	STRUCTURE
PHASE I *TAKE CONTROL*	3. Use a common language Challenge assumptions Understand what is being considered (situational awareness)	1. Know your users 2. Focus on user needs Remove bias and duplication Use appropriate methods	4. Think small (as in know the details)	Use a systematic mechanism of learning (bias towards data)		
PHASE II *GET FIT*	Be transparent (Bias towards open)	Focus on the outcome not a contract Think fast, inexpensive, restrained and elegant (FIRE) Use appropriate tools Be pragmatic Use standards where appropriate	Manage inertia Manage failure Effectiveness over efficiency	Bias towards action (learn by playing the game)	Move fast Strategy is iterative not linear	Think small (as in teams) Distribute power and decision making Think aptitude and attitude
PHASE III *BETTER WITH LESS*			Optimise flow (remove bottlenecks) Do better with less Set exceptional standards (great is just not good enough)	Bias towards the new (be curious, take appropriate risks)	Commit to the direction, be adaptive along the path Be the owner Think big, inspire others Embrace uncertainty Be humble (listen, be selfless, have fortitude)	Provide purpose, mastery & autonomy Seek the best
PHASE IV *REAL-TIME STRATEGY*				Listen to your ecosystems (future sensing engine)	Exploit the landscape There is no core (everything is transient)	There is no one culture Design for constant evolution

Break large landscapes into smaller projects in order to manage them better.

Don’t be put off by fears of increasing complexity.*

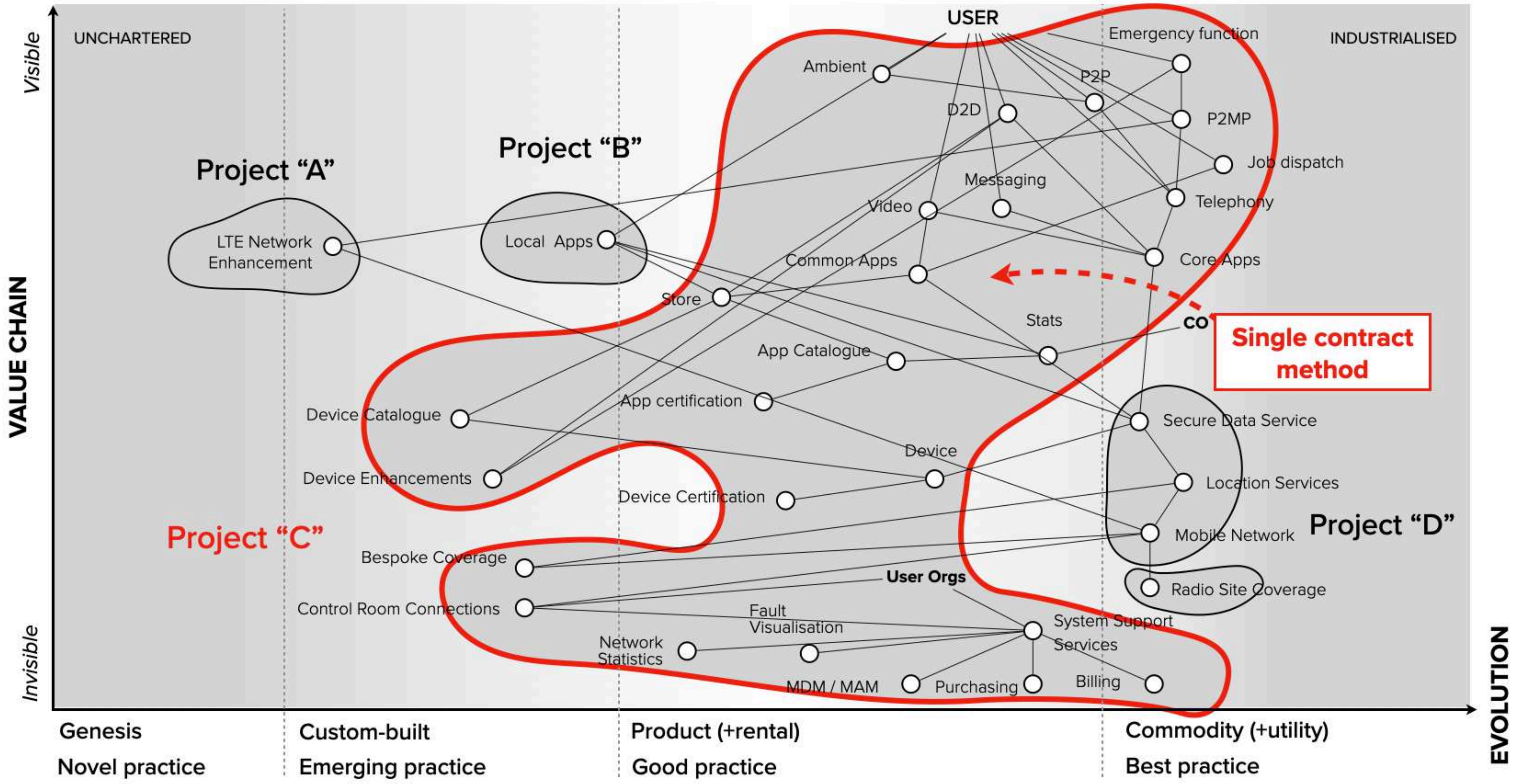


* That’s there already, just hidden, preventing you from managing it

Don’t do this!!

Group projects together based on financial or functional characteristics in order to make them more manageable.

This creates serious problems: Project “C” is too broad — with industrialised components (on the right) mixed with uncertain components (on the left).



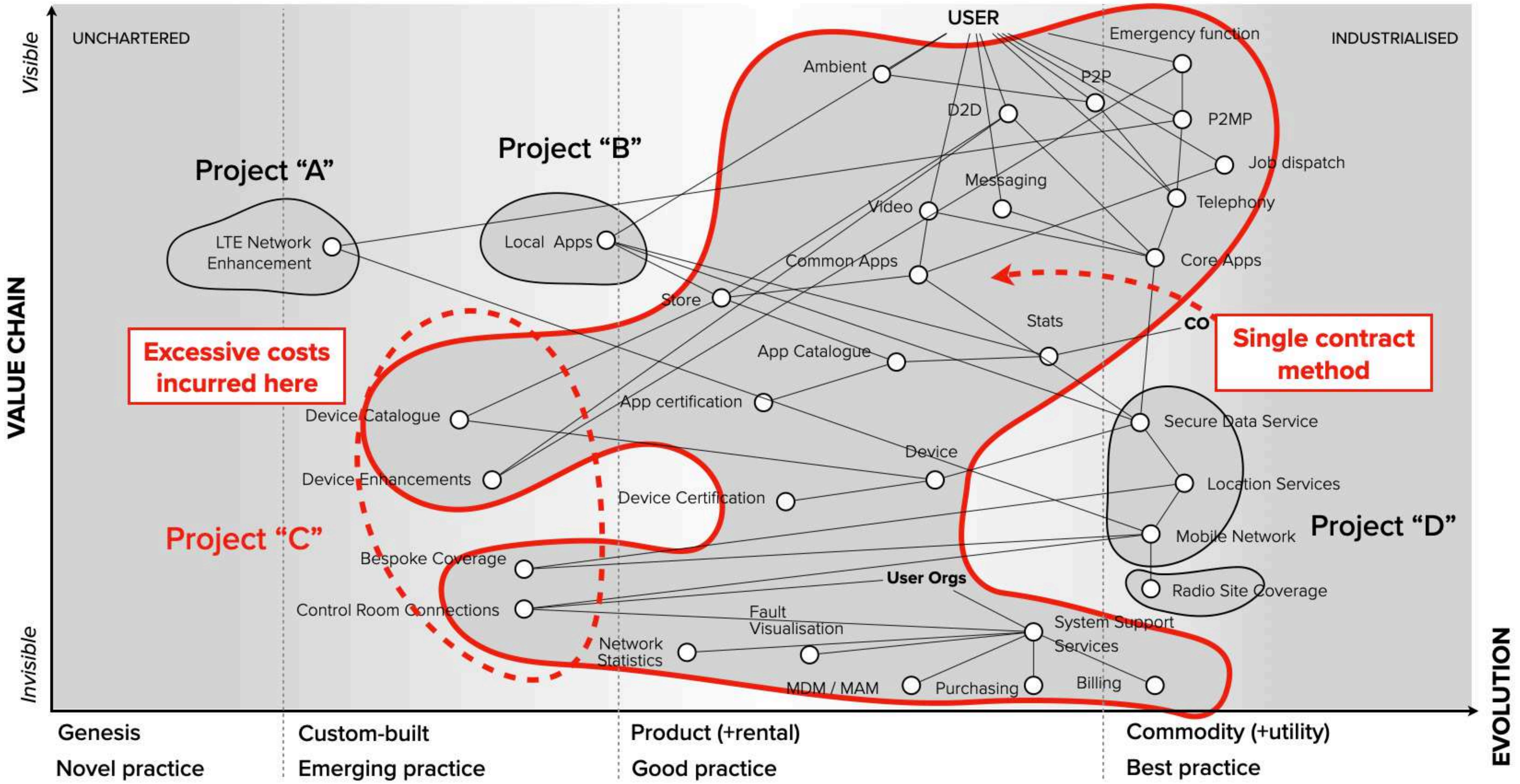
PowerMaps

Why it’s a problem

Components on the right of the map have a high-level of certainty. These can be delivered to specification.

Components on the left have a low-level of certainty. They will change often and incur excessive costs.

Project “C” will overrun its budgets and deadlines.



PowerMaps

The problem is “we want to know what is being delivered” from components where there is too much uncertainty.

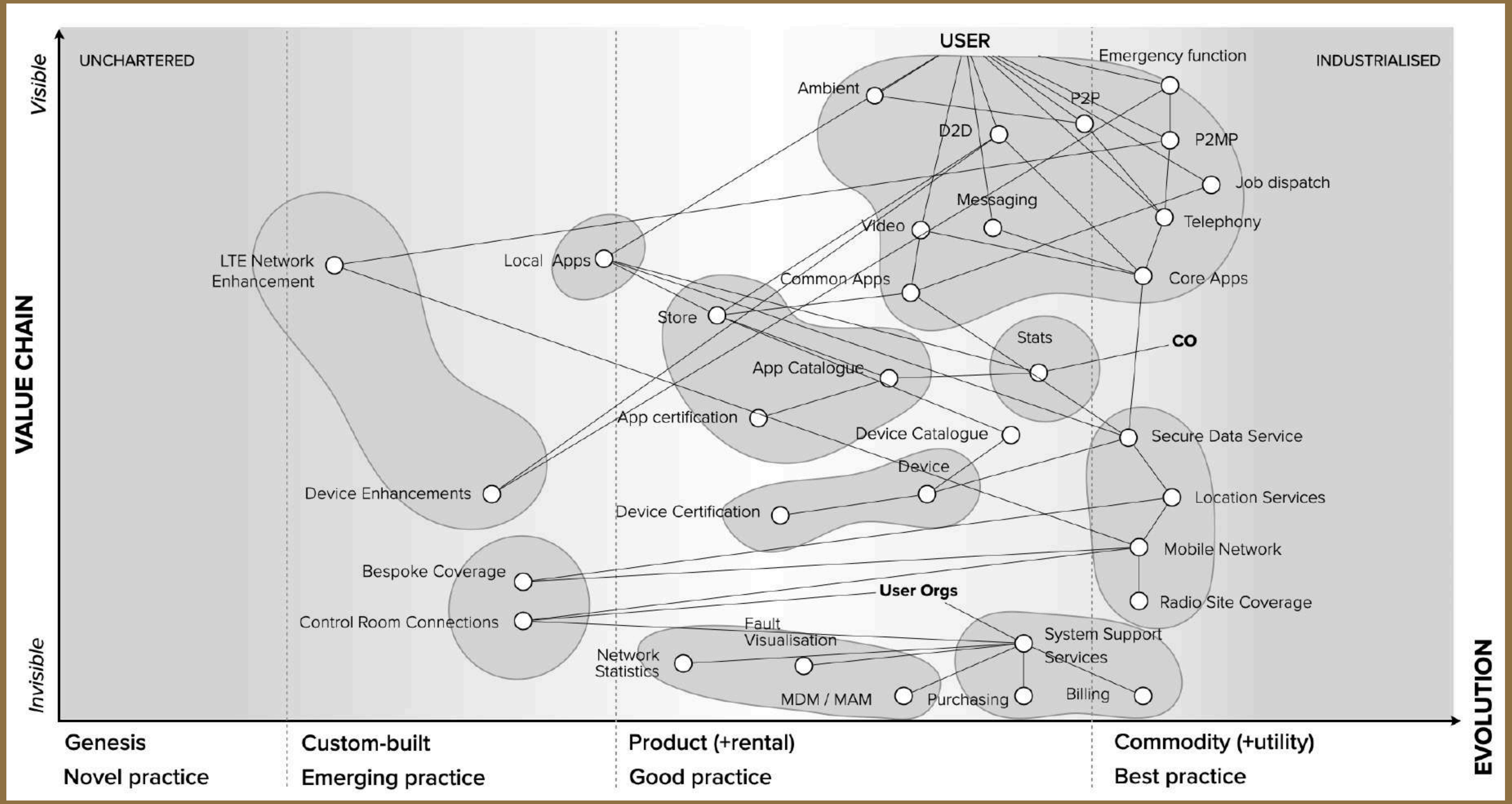
The only guarantee is that custom-built components will change, costs will spiral and disputes will break-out.

PowerMaps

Someone will suggest the solution is “better specification”.

This will only increase costs as you try (and fail) to define the uncertain.

The way to solve this problem is to map the Landscape then overlay your projects on it.



USER
UNCHARTERED
INDUSTRIALISED
Visible
Invisible
VALUE CHAIN
EVOLUTION
Emergency function
Ambient
P2P
D2D
P2MP
Job dispatch
Messaging
Video
Telephony
LTE Network Enhancement
Local Apps
Common Apps
Core Apps
Store
Stats
CO
App Catalogue
App certification
Device Catalogue
Secure Data Service
Device
Device Enhancements
Device Certification
Location Services
Mobile Network
Bespoke Coverage
User Orgs
Radio Site Coverage
Control Room Connections
Fault Visualisation
System Support Services
Network Statistics
MDM / MAM
Purchasing
Billing
Genesis
Novel practice
Custom-built
Emerging practice
Product (+rental)
Good practice
Commodity (+utility)
Best practice