Why, When, and How?

Why?

Why do we care about software architecture?

Proposition

Software architecture is about time, cost, and the tradeoffs between them.

Time tradeoffs

- Build time vs. lifespan
- Build time vs. change time
- Change time vs. lifespan
- Everything vs. cost

Discuss: tradeoffs you've made in the past. Decision, thought process, consequences.

Architecture guides

- Runtime interactions
- Division of labor
- Language

When?

When

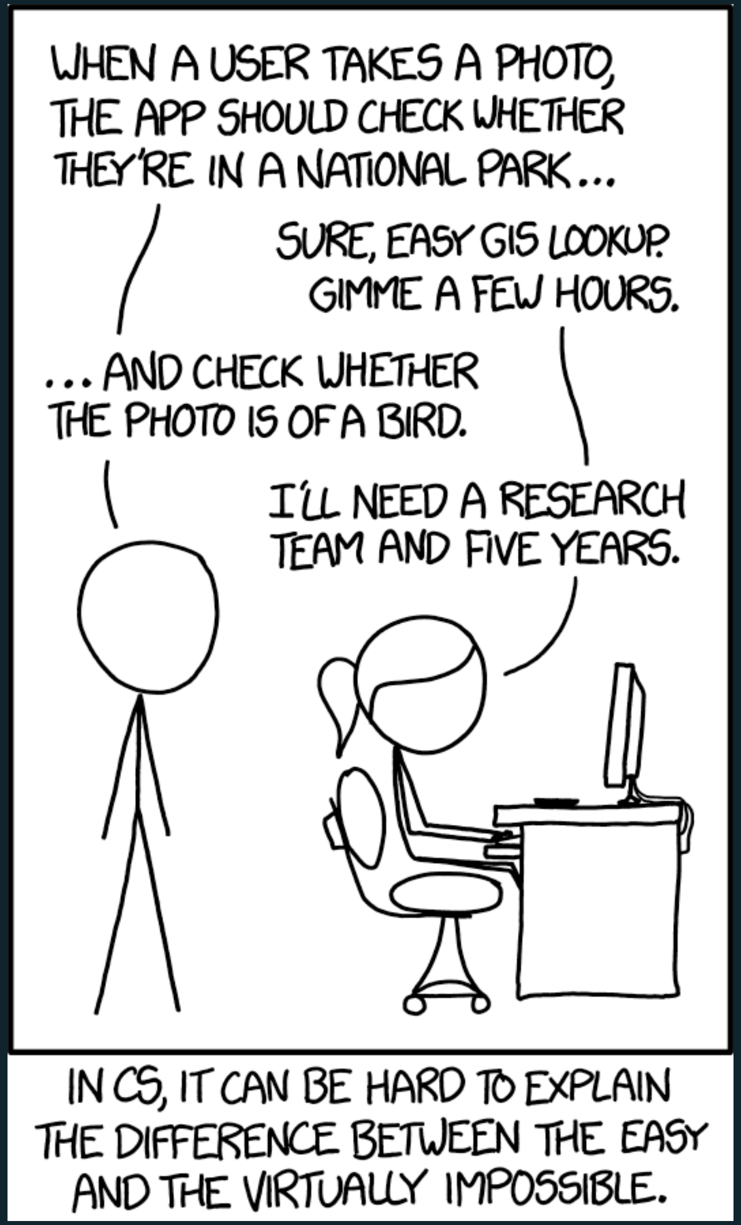
- Do we "do the architecture"?
- Is it finished?

Some up front

- Business Goals
- Constraints
- Architecture Quality Scenarios
- Architecturally Significant Requirements

Enough to find (most of) the "architecture killers"

An Architecture Killer


WHEN A USER TAKES A PHOTO, THE APP SHOULD CHECK WHETHER THEY'RE IN A NATIONAL PARK...
SURE, EASY GIS LOOKUP. GIMME A FEW HOURS.
...AND CHECK WHETHER THE PHOTO IS OF A BIRD.
I'LL NEED A RESEARCH TEAM AND FIVE YEARS.
IN CS, IT CAN BE HARD TO EXPLAIN THE DIFFERENCE BETWEEN THE EASY AND THE VIRTUALLY IMPOSSIBLE.

Business goals

Why does the system exist?

More revenue? Decrease cost?

For whom?

Constraints

Outside forces applied to project.

Break a contraint means game over. (Otherwise it's just a guideline.)

Stakeholders contribute constraints.

(But team's decisions can create future constraints.)

Architecture Qualities: Observed at runtime

- Performance
- Security
- Availability
- Usability

Architecture Qualities: Not observed at runtime

- Scalability
- Modifiability
- Portability
- Integrability
- Reusability
- Testability

Architecture Quality Scenarios

— Context
— Stimulus from a source
— Measurable response from a component

E.g.:

"When primary site stops responding, secondary site takes over front-end traffic in 5 minutes or less."

Base on project size and complexity

More early work:
- Large projects
- Complex
- Precise targets

"Off-the-shelf" architecture

— "It's a Ruby on Rails app"

Less attention required

Ongoing Work

- Take advantage of ambiguity.
- Prefer late binding.
- Create options.
- Last responsible moment.
- New discoveries or overlooked stakeholders

How?

Reminder

Principles and patterns about the interaction of parts within a system, and the orderly construction of that system.

Parts?

Development time	Deployment time	Runtime
Modules	Executables	Components
Libraries	Configurations	Processes
Artifacts		Roles

Interactions among them.

Search problem

Space of possible systems: infinite

Exclude parts of search space via:
- Constraints
- Architecture Quality Scenarios
- Architecturally Significant Requirements

Twin processes

1. Divergence: expand, explore, synthesize
2. Convergence: evaluate, eliminate, reconcile

Expand, explore, synthesize

Broaden range of considered structures.
- Bring in diverse ideas.
- Use architecture patterns.

Compare to past projects, personal experience, and public knowledge.

Push ideas further.

Generate new ideas via transforms: abstract, instantiate, split, substitute, augment.

Evaluate, eliminate, reconcile

Eliminate anything that breaks a constraint

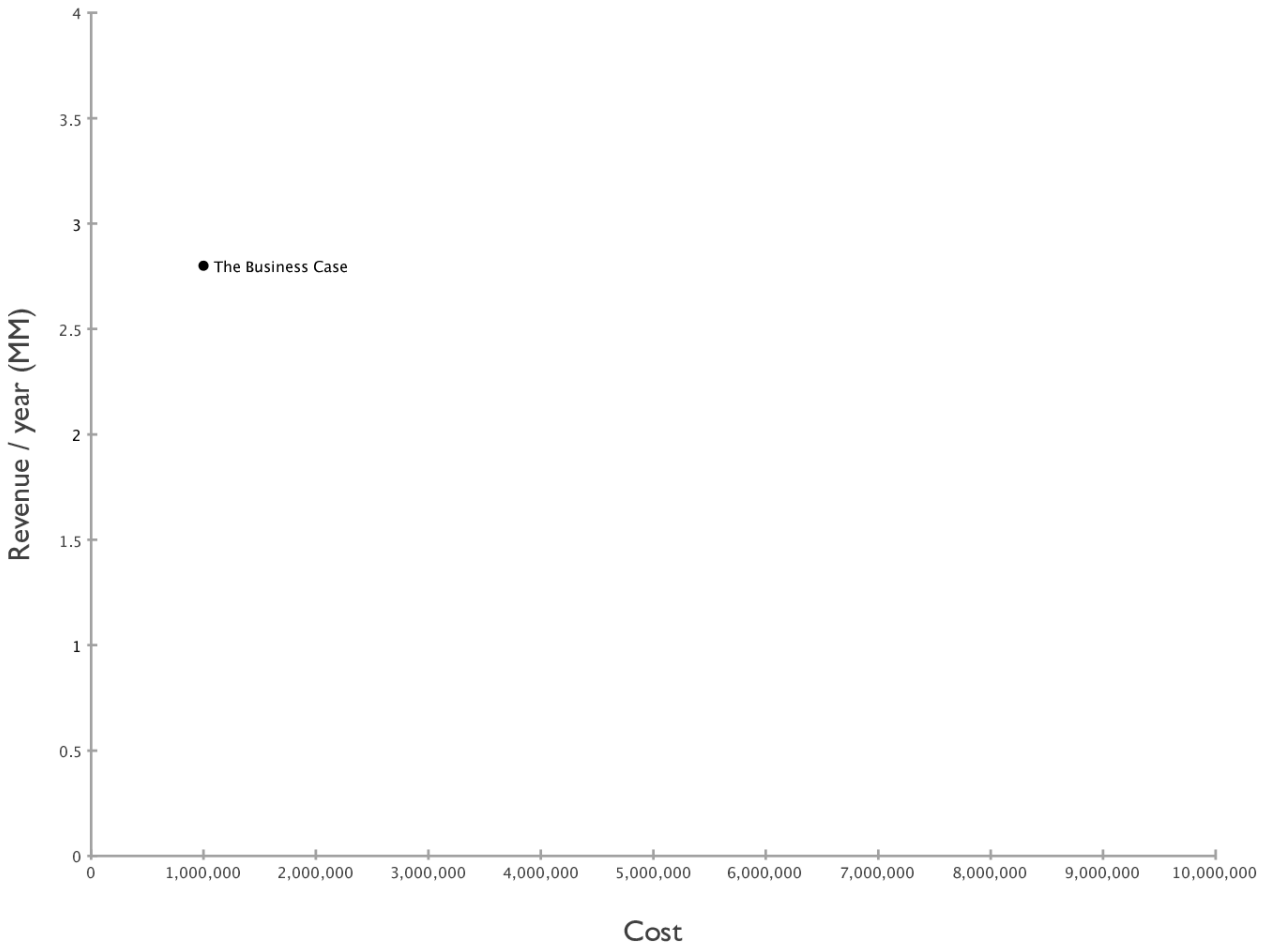
Test ideas by walking through scenarios & ASRs

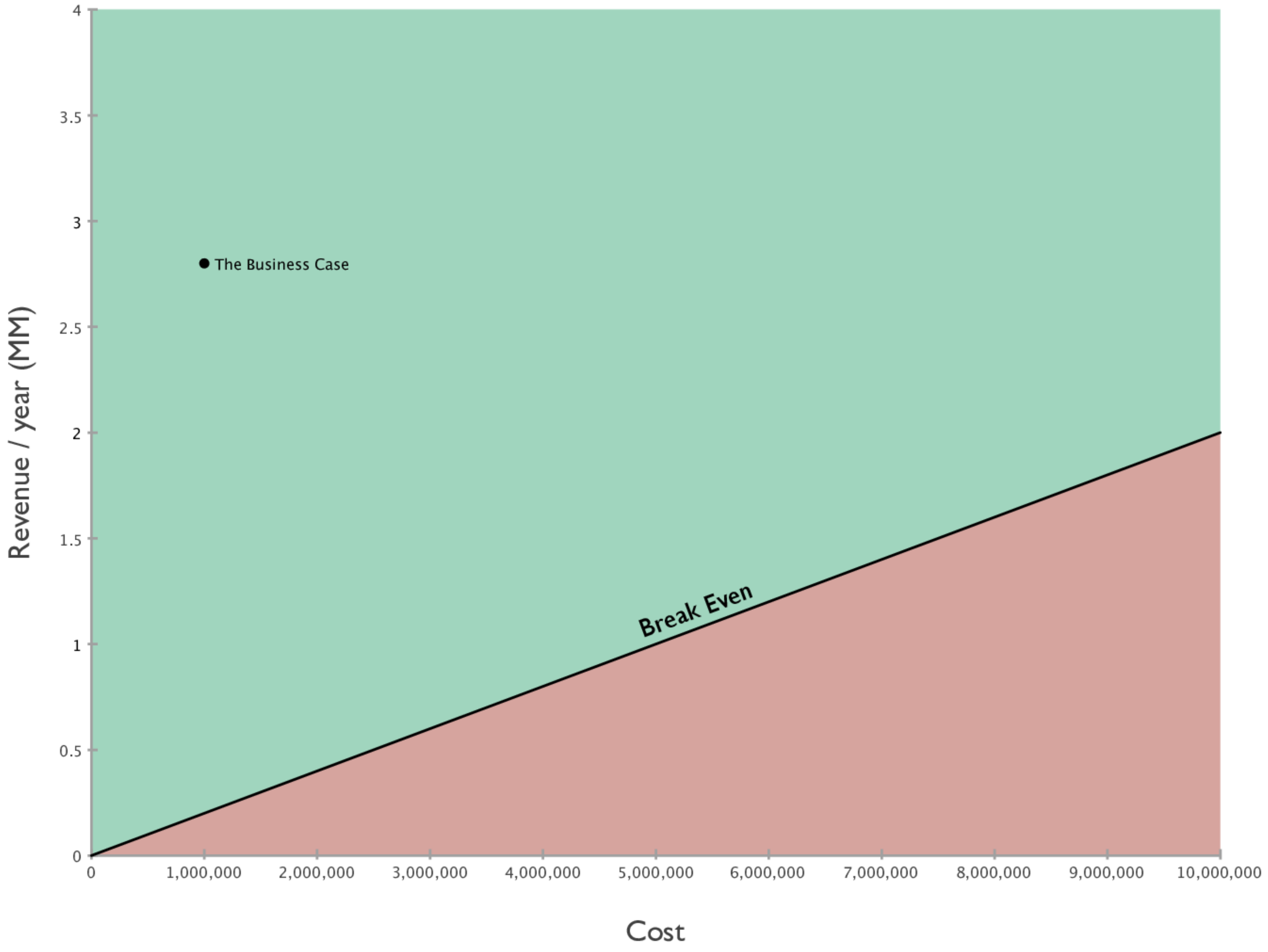
Compare competing ideas via architecture quality scenarios

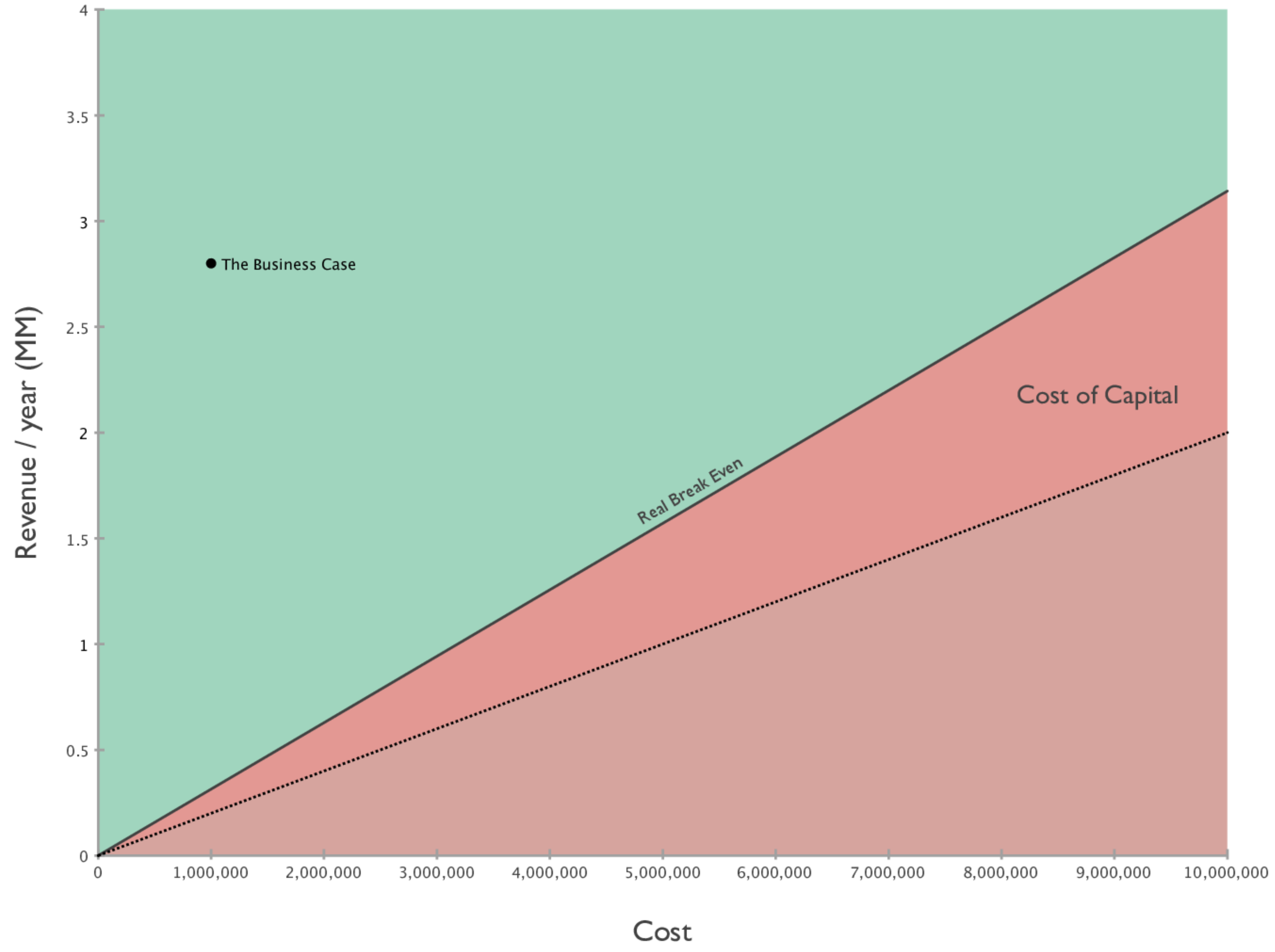
Consider construction time, system life span, change time

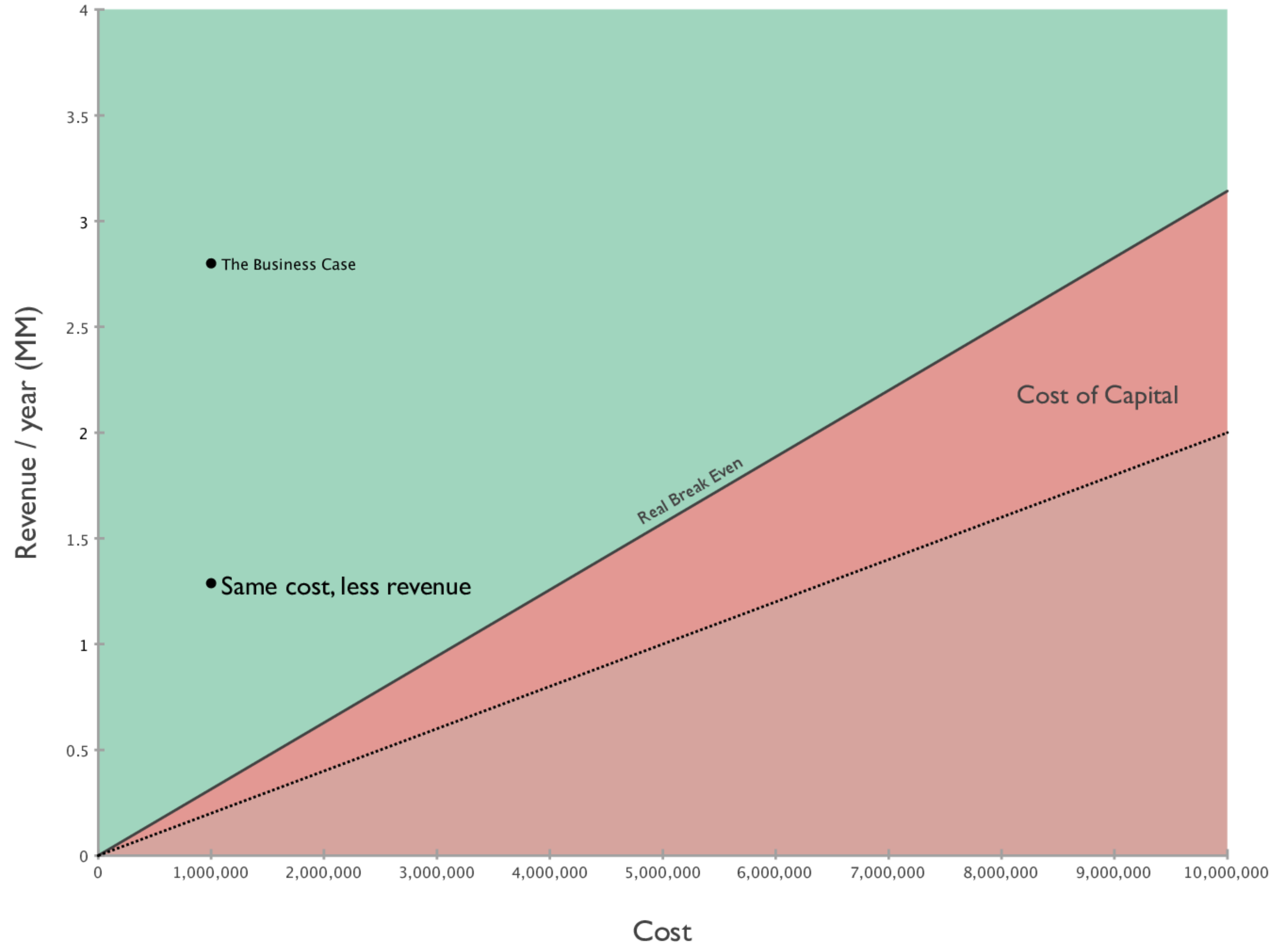
Uncertainty and Risk

All this happens in an environment of swirling change and uncertainty.

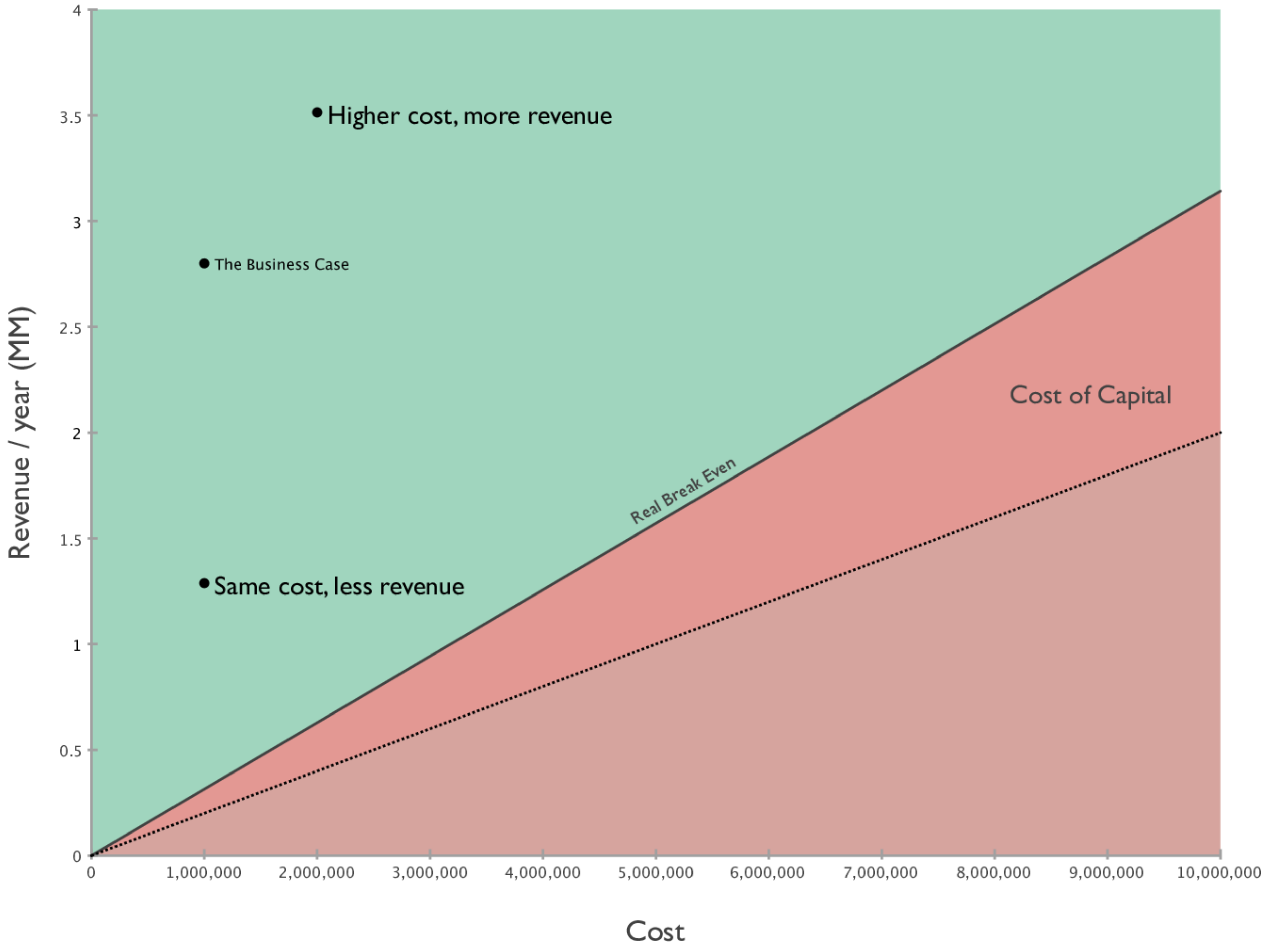

Revenue / year (MM)
4
3.5
3
2.5
2
1.5
1
0.5
0
The Business Case
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000
Cost


Revenue / year (MM)
4
3.5
3
2.5
2
1.5
1
0.5
0
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000
Cost
● The Business Case
Break Even


Revenue / year (MM)
4
3.5
3
2.5
2
1.5
1
0.5
0
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000
Cost
The Business Case
Real Break Even
Cost of Capital



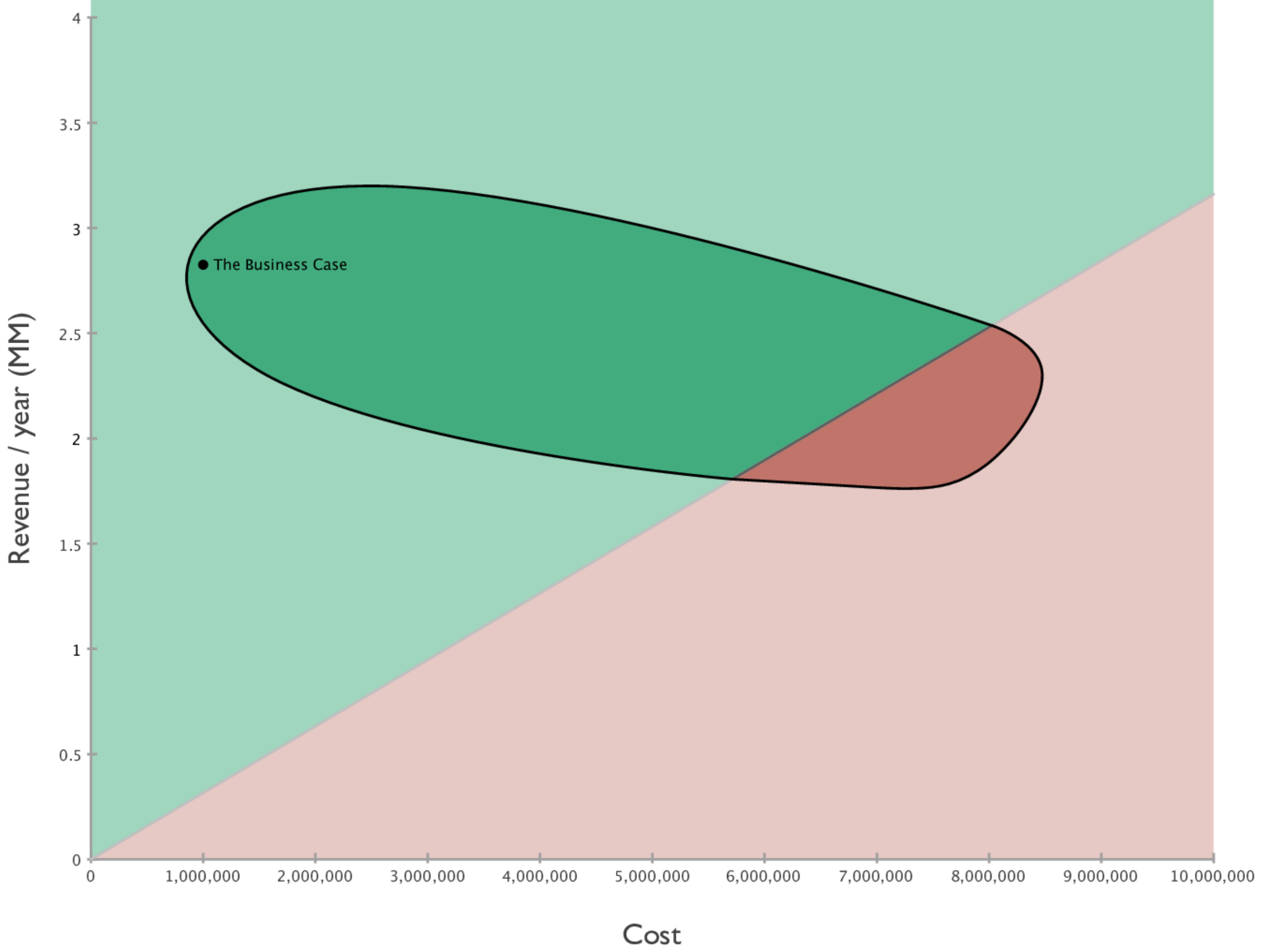
Revenue / year (MM)
4
3.5
3
2.5
2
1.5
1
0.5
0
The Business Case
Same cost, less revenue
Real Break Even
Cost of Capital
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000
Cost



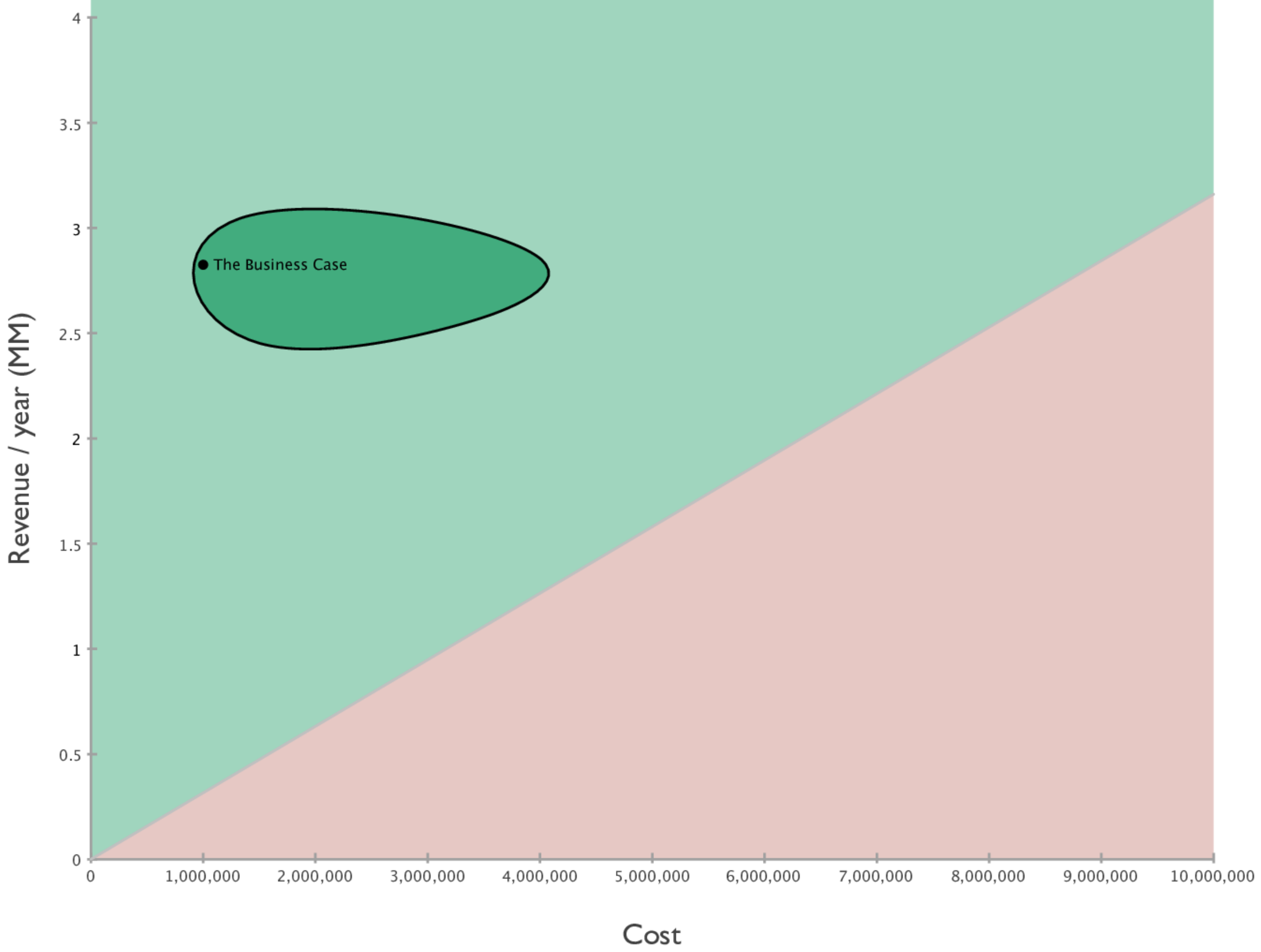
Revenue / year (MM)
Cost
Higher cost, more revenue
The Business Case
Same cost, less revenue
Real Break Even
Cost of Capital
4
3.5
3
2.5
2
1.5
1
0.5
0
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000

Economic Value of Information

- Measurements reduce uncertainty
- They also cost money
- Reduced uncertainty sometimes has value



Revenue / year (MM)
4
3.5
3
2.5
2
1.5
1
0.5
0
The Business Case
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000
Cost



Revenue / year (MM)
4
3.5
3
2.5
2
1.5
1
0.5
0
The Business Case
0
1,000,000
2,000,000
3,000,000
4,000,000
5,000,000
6,000,000
7,000,000
8,000,000
9,000,000
10,000,000
Cost