

INFO5992 EXAM CHEATSHEET

Understanding IT Innovations - Complete Reference

1. Innovation Types & Strategies

1.1 Disruptive Innovation

- **Low-End Disruption:** Targets overserved customers with “good enough” cheaper solutions → incumbents ignore
- **New-Market Disruption:** Creates new market serving non-consumers → different value proposition
- **Sustaining Innovation:** Improves existing products along established dimensions → incumbents excel

1.2 Innovation Dilemma

- Incumbents allocate resources to high-margin customers
- Ignore low-end threats until too late
- Existing capabilities become obsolete

1.3 Competence Types

- **Competence-Enhancing:** Builds on existing knowledge → incumbents have advantage
- **Competence-Destroying:** Requires new capabilities → startups have advantage (clean slate)

1.4 Technological Discontinuity

Definition: Dramatic breakthrough that makes existing tech obsolete

Why Companies Choose It:

- Overcome competitive stagnation
- Access new markets/capabilities
- Leapfrog competitors
- Respond to market shifts

Types:

- *Competence-enhancing:* Leverages existing skills
- *Competence-destroying:* Requires new expertise

Impact on Startups:

- Opportunity window for new entrants
- Incumbents' advantages neutralized
- Market uncertainty → experimentation phase

2. Dominant Design & Lifecycle

2.1 Dominant Design Phases

1. **Fluid Phase:** Multiple competing designs, high uncertainty, product performance focus
2. **Transitional Phase:** Design standards emerge, competition intensifies
3. **Specific Phase:** One design dominates, shift to process innovation & cost reduction

2.2 Why Dominant Design Matters

- Reduces uncertainty for customers & producers

- Enables economies of scale & network effects
- Compatibility with ecosystem/infrastructure
- Signals stability to mainstream market

2.3 When Dominant Design May NOT Emerge

- High customization needs (different segments need different designs)
- Rapid technological change prevents standardization
- Strong network effects lock in multiple competing standards
- Regulatory fragmentation across markets
- Low switching costs → continuous experimentation

2.4 Choosing Dominant Design (Startup)

- Customer acceptance & reduced adoption risk
- Compatibility with existing systems
- Cost & scalability for mass market
- Performance reliability benchmarks
- Network effects & ecosystem leverage

3. Technology Adoption Lifecycle

3.1 Adoption Segments

1. **Innovators** (2.5%): Tech enthusiasts, high risk tolerance
2. **Early Adopters** (13.5%): Visionaries, willing to experiment
3. **Early Majority** (34%): Pragmatists, need proven solutions
4. **Late Majority** (34%): Conservatives, adopt when necessary
5. **Laggards** (16%): Skeptics, resist change

3.2 Crossing the Chasm

- **The Gap:** Between Early Adopters & Early Majority
- **Why Hard:** Early majority demands reliability, references, established standards; startups lack credibility
- **How Dominant Design Helps:** Reduces uncertainty, signals stability, builds trust, enables ecosystem

4. Technology Hype Cycle

4.1 5 Stages

1. **Innovation Trigger:** Breakthrough sparks interest
2. **Peak of Inflated Expectations:** Unrealistic hype, many experiments
3. **Trough of Disillusionment:** Reality fails expectations, failures occur
4. **Slope of Enlightenment:** Practical applications emerge, *dominant design often emerges here*
5. **Plateau of Productivity:** Mainstream adoption, standards solidify

Relationship with Dominant Design: Competing designs

in early stages → dominant design stabilizes market during Slope/Plateau → reduces uncertainty → enables scale

5. Value Chain vs Value Network

5.1 Value Chain (Porter)

- **Linear model:** Sequential activities adding value
- **Primary activities:** Inbound logistics, operations, outbound logistics, marketing/sales, service
- **Support activities:** Infrastructure, HR, tech development, procurement
- **Focus:** Internal efficiency, cost reduction
- **Example:** Manufacturing (raw materials → production → distribution → retail)

5.2 Value Network

- **Network model:** Multiple players co-create value
- Interdependent relationships, not linear
- Value created through interactions & connections
- **Focus:** Ecosystem collaboration, network effects
- **Example:** Platform (app developers + users + payment providers all create value together)

5.3 Key Differences

Value Chain	Value Network
Linear flow	Multi-directional
Single firm focus	Ecosystem focus
Sequential activities	Simultaneous interactions
Efficiency-driven	Network effects-driven

6. Value Proposition Canvas

6.1 Customer Profile

- **Jobs:** Tasks customers accomplish (functional, social, emotional)
- **Pains:** Obstacles, risks, negative emotions
- **Gains:** Desired outcomes, benefits, aspirations

6.2 Value Map

- **Products/Services:** What you offer
- **Pain Relievers:** How you eliminate/reduce pains
- **Gain Creators:** How you create benefits

6.3 Achieving Fit

- Align value map with customer profile
- Address important jobs, alleviate major pains, create meaningful gains
- = **Product-Market Fit (PMF)**

6.4 Value Types

- **Functional:** Solves practical problem
- **Emotional:** Makes customer feel better

- **Social:** Status, belonging, impression
- **Life-Changing:** Transforms customer's life

7. Customer Development Process

7.1 4 Stages

1. **Customer Discovery:** Test problem hypotheses, identify real pains/needs, get out of building
2. **Customer Validation:** Test solution with MVP, achieve repeatable sales model
3. **Customer Creation:** Scale demand, execute marketing/sales
4. **Company Building:** Transition to execution-focused organization

7.2 Key Principles

- **Get Out of Building:** Talk to real customers, not desk research
- **Build-Measure-Learn:** Fast iteration cycles
- **Pivot or Persevere:** Change strategy based on validated learning
- **MVP:** Test assumptions with least effort

7.3 Creating Value in Customer Dev

- Align value map with customer profile through interviews
- Validate problem-solution fit before building
- Iterate based on feedback to strengthen value proposition
- Test with prototypes/wireframes in discovery phase
- Build high-fidelity MVP in validation phase
- Verify scalability & repeatability of business model

8. Lean Startup vs Waterfall

8.1 Why Waterfall Fails for Startups

- Linear & rigid → can't adapt to feedback
- Late customer involvement → build wrong product
- High waste if assumptions wrong
- Assumes clear requirements upfront (unrealistic for innovation)

8.2 Lean Advantages

- **Build-Measure-Learn loop:** Fast prototyping & testing
- Early customer involvement → validate assumptions
- Flexible pivots based on feedback
- Resource-efficient → focus on essential features
- Reduces risk through validated learning
- Supports experimentation & iteration

9. Open Innovation

9.1 3 Innovation Flows

1. **Outside-In:** Bring external ideas/tech into company (licensing in, partnerships, crowdsourcing)
2. **Inside-Out:** Commercialize internal ideas externally (spin-offs, licensing out, selling IP)
3. **Coupled:** Combine both through alliances, joint ventures, co-creation

9.2 Open Source vs Proprietary

Open Source	Proprietary
Community-driven innovation	Controlled development
Lower costs (no licensing)	Revenue from licenses
Transparency & auditability	Trade secrets protected
Faster iteration	Competitive advantage
Interoperability & standards	Lock-in strategy

9.3 Why Open Source Drives Innovation (4+ Reasons)

1. **Faster innovation:** Global community contributes improvements continuously
2. **Lower costs:** No licensing fees → allocate resources to differentiation
3. **Transparency:** Code auditability builds trust (critical for finance, healthcare)
4. **Interoperability:** Easier integration with existing systems via standards
5. **Flexibility:** Customize to specific needs without vendor lock-in
6. **Community support:** Faster bug fixes, security patches, feature development
7. **Talent attraction:** Developers prefer working with open technologies

9.4 Open Source Benefits for Startups

- **Standardization:** Compliance with industry norms (e.g., Linux, PostgreSQL)
- **Interoperability:** Easy integration with existing systems
- **Community innovation:** Continuous improvements (security, performance)
- **Lower costs:** Allocate resources to core differentiation
- **Trust:** Code auditability for regulatory compliance

9.5 Closed to Open Innovation Shift

Reasons for Shift:

- Access external expertise & reduce R&D costs
- Faster time-to-market through collaboration
- Tap into global talent pool
- Share development risk with ecosystem

Risks:

- Loss of IP control & competitive advantage
- Quality control challenges with external contributors
- Coordination overhead & integration complexity
- Risk of free-riding by competitors

10. APIs & Modularity

10.1 API Types

1. **API as Product:** Core business offering (Stripe, Twilio, AWS)
2. **API Enhancing:** Adds functionality to existing products (Google Maps API)
3. **API Promoting:** Drives adoption & ecosystem growth (Twitter API)

10.2 How APIs Enable Innovation

- **Modularity:** Decouple components → independent development
- **Scalability:** Add features without rebuilding core system
- **Third-party integration:** External developers extend functionality
- **Ecosystem growth:** Network effects as more developers build on platform
- **Faster iteration:** Update modules independently

10.3 Modularity in Software Architecture

- **Definition:** Breaking system into independent, interchangeable components
- **Benefits:**
 - Parallel development by different teams
 - Easier testing & debugging
 - Component reusability across projects
 - Lower coupling → changes don't cascade
 - Enable third-party innovation via APIs
- **Example:** Microservices architecture (authentication, payment, notification as separate modules)

11. Platform Business Models

11.1 Platform Ecosystem Model

- **Integrator Platform:** Mediates between external innovators & customers
- Controls transactions & interactions
- Benefits from network effects
- Monetizes through fees, subscriptions, data

11.2 Key Players in Platform

- **Producers:** Create value (app developers, content creators)
- **Consumers:** Use value (end users)
- **Platform Owner:** Provides infrastructure & governance
- **Complementors:** Enhance platform value (device makers, payment providers)
- **Regulators:** Ensure compliance & safety

11.3 Platform Types

1. **Transaction platforms:** Facilitate exchanges (Uber, Airbnb, eBay)
2. **Innovation platforms:** Enable third-party development (iOS, Android, AWS)
3. **Integrated platforms:** Combine both (Apple ecosystem)

11.4 Platform Monetization

- Transaction fees (commission on each sale)
- Subscription fees (premium features)
- Advertising revenue (targeted ads)
- Freemium model (free basic, paid premium)
- Data monetization (insights, analytics)

12. Crowdsourcing

12.1 4 Types

1. **Knowledge Discovery:** Tap distributed expertise (Innovenative - scientific problems)
2. **Broadcast Search:** Post problems, best solution wins (Kaggle - data science competitions)
3. **Peer-Vetted Creative:** Community evaluates ideas (Threadless - t-shirt designs)
4. **Distributed Human Intelligence Tasks:** Micro-tasks at scale (Amazon MTurk - labeling, surveys)

12.2 Challenges

- Quality control & reliability of contributions
- Intellectual property ownership disputes
- Participant motivation & retention
- Coordination costs & management overhead
- Free-riding & unequal contribution

12.3 Benefits

- Access to diverse expertise globally
- Cost-effective compared to in-house R&D
- Faster problem-solving through parallel efforts
- Identify innovative solutions from unexpected sources

13. Lead Users

13.1 Who They Are

- Experience needs ahead of market (future-oriented)
- High benefit from solutions (strong motivation)

- Often innovate solutions themselves (proactive)

13.2 Why Valuable for Sustainable Innovation

- Provide insights into future market needs
- Co-create products & provide early feedback
- Accelerate adoption as opinion leaders
- Reduce uncertainty & validate design for scalability
- Test extreme use cases that normal users won't encounter

13.3 Why Normal Users Less Effective (5 Reasons)

1. **Lack vision for future needs:** Focus on immediate problems, not emerging trends
2. **Limited technical expertise:** Can't suggest advanced/disruptive solutions → incremental feedback only
3. **Lower risk appetite:** Prefer proven solutions, avoid untested tech → slow radical innovation
4. **Incremental feedback only:** Minor improvements, not breakthrough ideas
5. **Biased by dominant designs:** Accustomed to current standards → resist creative thinking

14. Knowledge Sharing in Startups

14.1 How It Drives Innovation

- **Cross-pollination of ideas:** Different perspectives spark creativity
- **Faster problem-solving:** Collective intelligence reduces bottlenecks
- **Reduced duplication:** Share learnings to avoid repeated mistakes
- **Skill development:** Team learns from each other
- **Organizational learning:** Capture & reuse tacit knowledge

14.2 Mechanisms/Practices for Knowledge Sharing

1. **Cross-functional teams:** Break silos, share domain expertise
2. **Regular knowledge-sharing sessions:** Sprint demos, brown bags, tech talks
3. **Documentation culture:** Wikis, runbooks, design docs
4. **Mentorship & pair programming:** Tacit knowledge transfer
5. **Retrospectives:** Capture lessons learned from projects
6. **Internal communities of practice:** Groups focused on specific domains
7. **Open communication tools:** Slack channels, forums for async sharing

15. Coopetition

15.1 Definition

Collaboration with competitors instead of pure disruption

15.2 Why Startups Choose It

- **High entry barriers:** Costly infrastructure, regulations → partnerships needed
- **Shared resources:** Access to distribution, tech, customer base
- **Network effects:** Faster adoption & ecosystem growth through collaboration
- **Risk reduction:** Lower financial & market uncertainty
- **Complementary strengths:** Startup innovation + incumbent scale/credibility

15.3 Examples

- Fintech startups partnering with banks (access to customers + regulatory compliance)
- EV startups using established automakers' charging networks
- SaaS companies integrating with competitors to serve customers better

16. Organizational Structures

16.1 Mechanistic Structure

- High formalization, centralized decisions
- Rigid hierarchy, clearly defined roles
- Efficient for stability, routine tasks
- Limits creativity & flexibility

16.2 Organic Structure

- Low formalization, decentralized decisions
- Flexible processes, collaborative
- Encourages adaptability & innovation

16.3 Making Mechanistic Innovative

1. **Cross-functional teams:** Encourage collaboration across departments
2. **Reduce formalization:** Allow flexibility in processes
3. **Promote risk-taking:** Failure viewed as learning, not punishment
4. **Allocate innovation time:** e.g., Google's "20% time"
5. **Reward creativity:** Incentives for innovative ideas
6. **Skunk Works teams:** Small, autonomous groups for disruptive projects
7. **Flatten hierarchy:** Reduce approval layers for faster decisions

17. Startup Failure Factors

17.1 Top 3 Reasons

1. **Lack of Product-Market Fit:** Product doesn't solve real customer needs → poor adoption. Built without validating demand.

- Insufficient Capital:** Run out of cash before achieving scale. High burn rates, long R&D cycles (hardware/deep-tech).
- Weak Business Model:** No clear revenue model, mispricing, unclear monetization → unsustainable operations.

17.2 Other Contributing Factors

- Poor team dynamics & founder conflicts
- Inability to pivot when assumptions fail
- Intense competitive pressure
- Premature scaling before PMF
- Ignoring customer feedback

18 18. Business Model Canvas

18.1 Key Components

Customer Segments: Who are you serving?

- Mass market, niche, segmented, diversified, multi-sided

Value Propositions: What value do you deliver?

- Newness, performance, customization, design, brand, price, convenience, risk reduction

Key Resources: What assets required?

- Tangible:** Physical (facilities, equipment), Financial (cash, credit)
- Intangible:** Intellectual (patents, IP, data), Human (skills, expertise, talent)

Key Activities: What do you do?

- Production, problem-solving, platform/network management

Revenue Streams: How do you make money?

- Asset sale, usage fee, subscription, licensing, advertising, freemium

18.2 Why Customer Segments & Value Props Critical

- Focus resources:** Target right customers, avoid wasting effort
- Achieve PMF:** Clear value prop aligned with segment needs
- Differentiation:** Stand out from competitors in specific segments
- Scalability:** Repeatable model for similar customers
- Revenue model:** Pricing & monetization fit customer willingness to pay

19 19. Diffusion of Innovation

19.1 Diffusion Strategies for Tech Startups

- Target early adopters:** Identify visionaries willing to take risks, use as reference customers
- Build credibility:** Case studies, testimonials, proof of

concept with reputable customers

- Reduce adoption barriers:** Free trials, freemium model, easy onboarding
- Leverage network effects:** Incentivize referrals, viral loops, community building
- Partner with complementors:** Integrate with established platforms/ecosystems
- Educate market:** Content marketing, webinars, thought leadership to build awareness
- Align with dominant design:** Reduce perceived risk for mainstream adopters

19.2 Barriers to Innovation Implementation

- Organizational inertia:** Resistance to change, “not invented here” syndrome, fear of disrupting existing business
- Resource constraints:** Insufficient budget, talent shortage, competing priorities
- Lack of leadership support:** Innovation not prioritized, risk-averse culture
- Poor communication:** Silos prevent knowledge sharing, misaligned incentives

20 20. Agentic AI vs Autonomous

20.1 Agentic AI

- Proactive & goal-driven:** Pursues objectives independently
- Adaptive reasoning:** Adjusts strategy based on environment
- Multi-step planning:** Breaks down complex tasks
- Example:** AI research assistant formulating hypotheses & designing experiments

20.2 Autonomous Systems

- Reactive & task-specific:** Executes predefined tasks
- Rule-based:** Follows programmed instructions
- Limited adaptability:** Within narrow scope
- Example:** Self-driving car following traffic rules

21 21. General Purpose Technology

21.1 Characteristics of GPT

- Wide applicability across sectors
- Continuous improvement over time
- Spawns complementary innovations
- Transforms entire economies
- Examples:** Electricity, internet, steam engine

21.2 Can Generative AI be GPT?

YES - Supporting Arguments:

- Wide applicability:** Content creation, coding, design,

research, customer service across all industries

- Continuous improvement:** Rapid iteration, models improving exponentially
- Complementary innovations:** New tools, applications, business models emerging
- Productivity transformation:** Automating knowledge work at scale

Counterarguments (if needed):

- Still early stage, adoption not yet universal
- Economic transformation impact unclear

22 22. EV Types (if relevant)

22.1 Battery Electric Vehicle (BEV)

- 100% electric, no combustion engine
- Zero tailpipe emissions
- Example:** Tesla Model 3, Nissan Leaf

22.2 Plug-in Hybrid (PHEV)

- Electric motor + combustion engine
- Can run on electric only (limited range), then switches to gas
- Example:** Toyota Prius Prime, Chevrolet Volt

22.3 Range-Extended EV (REEV)

- Primarily electric, small engine as generator (doesn't drive wheels)
- Extends range when battery depleted
- Example:** BMW i3 with range extender

23 23. Exam Answer Templates

23.1 2-Mark Question (2-3 lines)

- Define concept in 1 line
- Key reason/application in 1-2 lines
- Example:** “PMF is degree to which product satisfies market demand. Value Prop Canvas achieves it by aligning customer jobs/pains/gains with pain relievers & gain creators.”

23.2 5-Mark Question

- 2-3 key points with brief explanation
- Each point: 1-2 lines
- Use bullet format
- Example:** See Section 1 questions in answer key

23.3 10-Mark Scenario

- Apply framework clearly
- 4-5 bullet points
- Each point: claim + reasoning + scenario application
- Use keywords marker looks for
- Example:** See Section 2 & 3 questions in answer key

24. Key Exam Keywords

24.1 Innovation Type

disruptive, sustaining, low-end, new-market, competence-destroying, competence-enhancing, radical, incremental

24.2 Adoption

chasm, early adopters, mainstream, pragmatists, proven solution, references, credibility

24.3 Design

dominant design, standardization, compatibility, network effects, economies of scale, interoperability

24.4 Customer Dev

MVP, pivot, validated learning, get out of building, problem-solution fit, PMF, iteration

24.5 Value Prop

jobs-pains-gains, pain relievers, gain creators, functional/emotional/social value, fit

24.6 Lean

Build-Measure-Learn, iterate, waste reduction, customer feedback loops, fast experimentation

24.7 Open Innovation

outside-in, inside-out, coupled, ecosystem, interoperability, collaboration

24.8 Platform

network effects, multi-sided, transaction fees, ecosystem, complementors

25. Quick Reference Frameworks

25.1 When Analyzing Innovation Type

1. Does it target overserved customers with cheaper solution? → *Low-end disruption*
2. Does it create new market for non-consumers? → *New-market disruption*
3. Does it improve along existing dimensions? → *Sustaining*
4. Does it make existing capabilities obsolete? → *Competence-destroying*

25.2 When Analyzing Platform Ecosystem

1. Identify key players (producers, consumers, complements)
2. Show mediation role of platform
3. Explain network effects
4. Discuss monetization strategy

25.3 When Analyzing Customer Development

1. Discovery: What problem? Get out of building, test hypotheses
2. Validation: Does solution work? MVP, test sell, positioning
3. Pivot considerations: What's wrong? How to address?
4. Use Build-Measure-Learn terminology

25.4 When Analyzing Value Proposition

1. Customer jobs (what trying to do?)
2. Pains (what frustrations?)
3. Gains (what benefits desired?)
4. How product addresses each

26. Common Question Patterns

26.1 "What type of innovation is X?"

- State innovation type clearly
- Provide 2 reasons with explanation
- Reference scenario specifics

26.2 "How can X help achieve Y?"

- Briefly define X
- List mechanisms/ways (3-4 points)
- Connect to outcome Y

26.3 "Why do startups fail / choose competition / etc?"

- List 3-5 clear reasons
- Each reason: 1-2 line explanation
- Use scenario context if given

26.4 "Explain difference between X and Y"

- Define both briefly
- Highlight 3-4 key differences
- Use table if helpful
- Provide example for each