

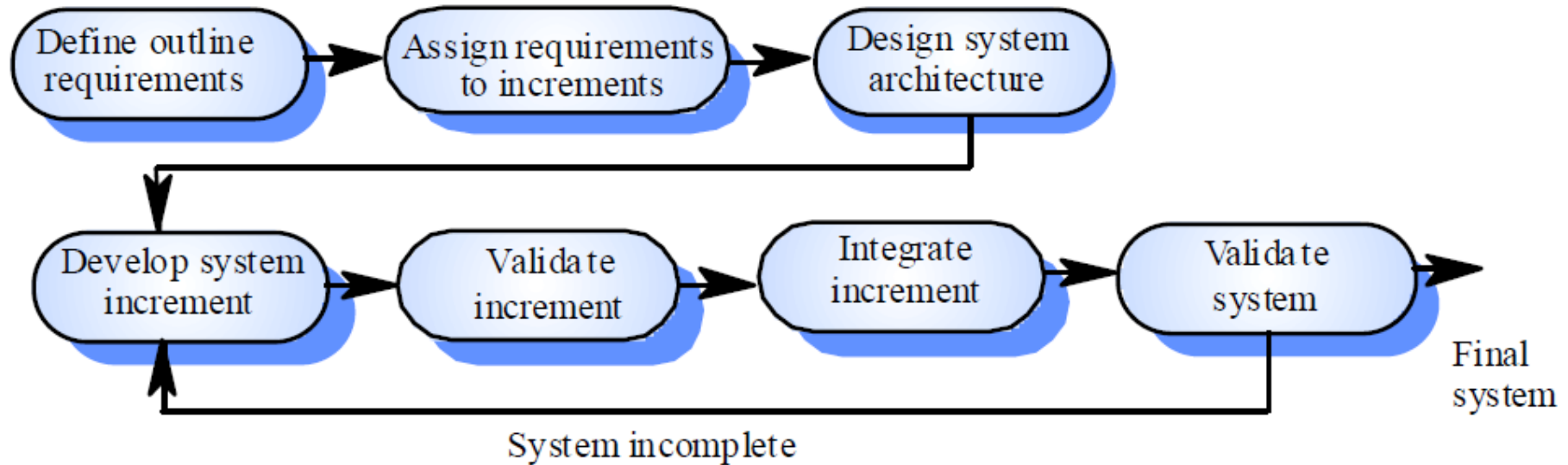
프로세스 반복

- 시스템 요구사항은 프로젝트가 진행되는 동안 항상 진화한다. 따라서 초기 단계를 재작업하는 프로세스 반복은 항상 대형 시스템을 위한 프로세스의 일부분이다.
- 반복은 일반적인 프로세스 모델의 어떤 것에도 적용될 수 있다.
- 2개의 (관련된) 접근법
 - 점진적(Incremental) 개발
 - 나선형(Spiral) 개발

점진적 개발

- 시스템을 한번에 인도하는 것이 아니라, 개발과 인도가 여러 번으로 나누어진다. 각각의 증가분에서는 요구되는 기능들을 인도한다.
- 사용자의 요구사항들은 우선순위가 매겨지며, 높은 우선순위를 갖는 요구사항들을 먼저 인도한다.
- 증가분의 개발이 시작되면, 요구사항에는 변동이 없게되며, 다음번 증가분에 대한 요구사항이 계속해서 진화하게 된다.

점진적 개발



점진적 개발의 이점

- 고객에게 필요한 부분이 각각의 증가분마다 인도될 수 있다. 따라서 시스템의 기능이 초기에 이용 가능하다.
- 초기의 증가분이 나중의 증가분에 대한 요구사항을 이끌어내는 프로토타입의 역할을 한다.
- 전체적인 프로젝트 실패에 대한 위험이 적음.
- 가장 높은 우선순위를 갖는 시스템 서비스는 많은 테스트를 거치게 된다.

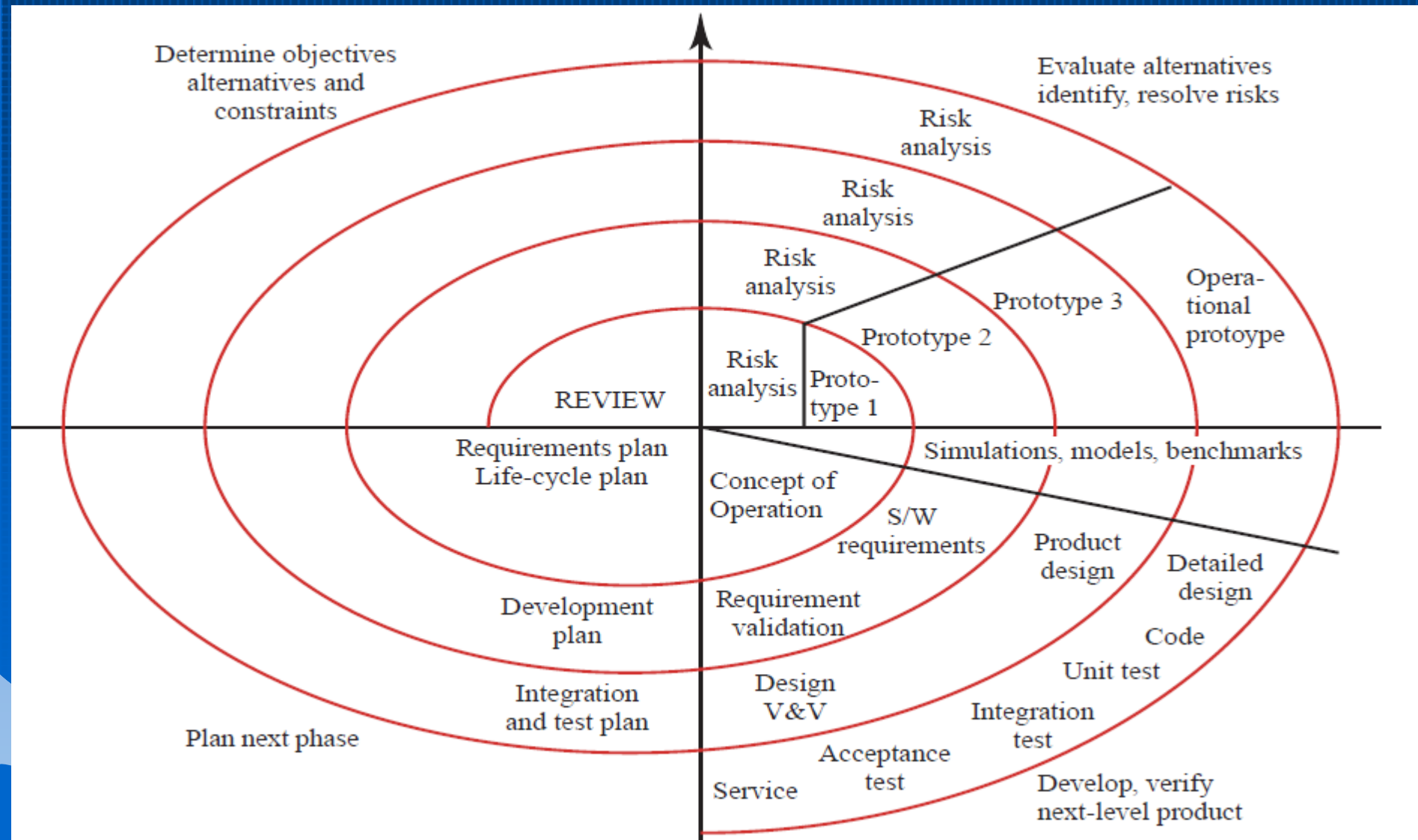
극단적(*Extreme*) 프로그래밍

- 매우 적은 기능성을 가진 증가분의 개발과 인도를 기초로 하는 개발에 대한 새로운 접근법
- 계속적인 코드 향상, 개발 팀에 사용자의 포함, 그리고 pairwise 프로그래밍에 의존함.

나선형(spiral) 개발

- 프로세스를 백트래킹을 가진 활동들의 순서가 아닌 나선형으로 표현.
- 나선내의 각각의 루프는 프로세스내의 단계를 나타냄.
- 프로세스 내내 위험(Risk)이 명백하게 측정되고 해결된다.

소프트웨어 프로세스의 나선형 모델



나선형 모델의 섹터

■ 목표 설정

- 프로젝트 단계를 위한 특정한 목적을 식별함.

■ 위험 측정과 감소

- 위험들을 측정하고, 주요한 위험들을 줄이기 위한 활동들이 위치함.

■ 개발과 확인

- 시스템에 대한 개발 모델을 선택. 일반적인 모델 중 어떤 것이라도 상관 없음.

■ 계획수립

- 프로젝트 검토, 나선형의 다음 단계에 대한 계획.