

# Orion

Mamoon Umar (22819222)

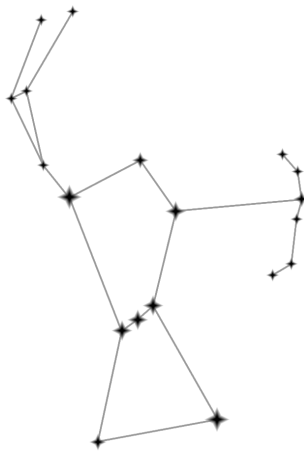
BSc Software Engineering  
University of Brighton

Saeed Malekeshahi Gheytaasi (Supervisor)

2025 – 2026

## Abstract

This report focuses on the design and implementation of a time-shared, distributed operating system. The system is built on the principles of Plan 9 by Bell Labs, emphasizing simplicity, modularity, and network transparency. Previous work in distributed systems has been researched and discussed. Orion implements these philosophies to create a lightweight system which allows users to seamlessly access resources across a network.



# 1 Project Definition

## 1.1 Scope

This project focuses on the design and implementation of Orion, a minimal distributed operating system developed to evaluate the applicability of Plan 9 principles in a modern context. The scope is limited to core operating system concerns such as namespace management, inter-process communication, resource management and distributed interaction between host and terminals; this prioritizes architectural and abstraction clarity over performance and stability. Large scale deployment, hardware support, and benchmarking are explicitly out of scope and will not be explored in this research paper. Qualitative comparisons against modern systems ensure a focus on structural complexity and abstraction models rather than empirical metrics. As a result, Orion should be treated as a research artifact and evaluative instrument to distill the research conducted on Plan 9 and distributed systems.

## 1.2 Objectives & Hypotheses

Building on the defined scope, the objectives of this project originated from the hypotheses. I anticipate practical outcomes from applying Plan 9 principles to modern distributed systems. By explicitly testing these hypotheses, Orion will serve as a platform for analysing design trade-offs between Plan 9 mechanisms and modern alternatives.

$\mathcal{H}_1$  Plan 9 design principles can be successfully applied to construct a functional, minimal distributed system in a modern context.

$\mathcal{H}_2$  Plan 9 abstractions reduce conceptual and implementation complexity compared to typical modern distributed system designs.

## 1.3 Methodology

### Project Management.

**Testing.** Conduct a set of tasks which exercise core Plan 9 mechanisms such as namespace management, inter-process communication, and efficient distributed resource access. Test over a network of machines to simulate distributed system interactions. Observe functional correctness, Plan 9 adherence, and clear design patterns.

## 2 Plan 9

### 2.1 Computing System History

**The fall of Multics.** Slow and inevitable, it was described by Dennis Ritchie.

The cessation of Bell Labs from the Multics project brought K. Thompson, D. Ritchie, M.D. McIlroy and J.F. Ossana to begin work on a new operating system. The group enjoyed the convenience of working with MIT and General Electric, yet placed greater importance on close communication and less complexity. [3]

**The rise of UNIX.** In 1969, the Bell Labs team began researching how to implement a much smaller operating system based on Multics' ideas, but able to run on the much smaller PDP-7 minicomputer. The following decade saw the creation of the C programming language by Ritchie, which then led to the birth of UNIX. It was adopted early by universities, research labs, and eventually industry. [1]

**Sunrise on Plan 9.** The inventors and now veterans of UNIX and the C language saw the success of their system whilst elsewhere, people slowly grew tired of centralized timesharing machines. The age of personal computers with microprocessors had arrived. The team from Bell Labs elaborated:

“Timesharing centralized the management and amortization of costs and resources; personal computing fractured, democratized, and ultimately amplified administrative problems. The choice of an old timesharing operating system to run those personal machines—” [2]

This evaluation of monolithic timeshared systems (UNIX) implemented on personal machines became the foundation of Plan 9.

### 2.2 Plan 9 Technology

#### 2.2.1 Design

”Build a UNIX out of a lot of little systems, not a system out of a lot of little UNIXes”  
— Bell Labs[2]

The first operating systems that ran on these modern computers were UNIX. However, as described by the creators, UNIX has trouble adapting to ideas invented after it. This led to poor implementations of key features such as networking and graphics. Bell Labs wanted the central administration of timeshared systems whilst also adopting modern personal computers to create a distributed OS for their own designs. [2]

Plan 9 has three main philosophies which are strictly adhered to for stable and reliable development:

1. Resources are named and accessed like files in a hierarchical system
2. There is a standard protocol for accessing these resources (in Plan 9 it is called 9P)
3. The disjoint hierarchies provided by different services are joined in a single private hierarchical file namespace

### 3 Distributed Systems

Present your results.

# Bibliography

- [1] Nokia. *50 Years of UNIX*. 2025. URL: <https://www.nokia.com/bell-labs/about/history/innovation-stories/50-years-unix/>.
- [2] Rob Pike et al. “Plan 9 from bell labs”. In: *Proceedings of the summer 1990 UKUUG Conference*. London, UK. 1990, pp. 1–9.
- [3] Dennis M. Ritchie. “The Evolution of the Unix Time-sharing System\*”. In: (1973). URL: <https://www.nokia.com/bell-labs/about/dennis-m-ritchie/hist.pdf>.