Networking China: The Grand Canal's Legacy

Preparation

- Number of words: 3949
- Group number: 9
- Group member: Keyi Heng, Yiduo Yun, Jialei Guo
- Demonstration Video link

Table of contents

Task Name	Major Contributors	Additional Contributors	Relevant Chapters in Report
Overview	Keyi Heng, Jialei Guo		1.1, 2.4
Technology	Keyi Heng		1.2, 2.1, 2.4
Commerce	Yiduo Yun		1.2, 2.2, 2.4
Culture	Jialei Guo		1.2, 2.3, 2.4
Future	Yiduo Yun	Keyi Heng	1.2, 2.4

Project Output	Output Description
Project Output Files	Zip File on Moodle
Project Website	https://github.com/JialeiGuo1108/The-Grand-Canal

1 Webpage Theme and Design

1.1 Webpage Theme

The Grand Canal is not just a river—it's a network. Over thousands of years, it enabled the spread of technology, the rise of trade, and the blending of cultures across China. Our project, "Networking China: The Grand Canal's Legacy," explores how this remarkable system connected people, ideas, and goods over time.

This project explores the Grand Canal not just as a historical subject, but as a data-rich system. Through the lens of temporal and spatial visualization, we trace how connections unfolded — across dynasties, across regions, across fields of human activity.

The Grand Canal of China is the world's earliest, longest, and largest artificial canal system, spanning a total length of 3200 kilometres and covering eight provincial-level administrative regions. Its history can be traced back to about 400 BC, representing the achievements of China's ancient water conservancy and shipping engineering technology, and it is an irreplaceable production, living, and ecological infrastructure (Jianmei, Mengqi & Jia 2022). For thousands of years, it has consistently promoted exchanges and communication among regions, which is of great significance. In 2014, the Grand Canal was designated a UNESCO World Heritage Site (Yan 2021).

By combining historical texts, technological diffusion records, and geospatial datasets, we reconstruct the evolving network of the Canal. Interactive maps, timelines, and cultural overlays enable users to see patterns, spot transitions, and uncover relationships hidden in the flow of time.

From the perspectives of digitalization and networking, this webpage attempts to answer the following core questions:

How did the Grand Canal serve as a connecting system to promote the development of Chinese history?

In fields such as technology, business, and culture, in which specific paths and nodes does its "network role" manifest?

How can we make the "fluidity" and "systematicness" of history visible and perceptible by leveraging data and visualization tools?

We believe that the digital humanities approach can not only reconstruct the multi-dimensional appearance of the Grand Canal, but also inspire us to reunderstand the proposition of "connection in civilization".

1.2 Webpage Design

1.2.1 Overall Design Philosophy and Style Positioning

The overall design of this webpage takes "the network nature of the Grand Canal" as its core, integrating historical narratives and data visualization, and emphasizing "the evolution process of the Grand Canal as a spatio-temporal network". We adopt the three major theme lines of "technology - Business - culture", with time as the axis and space as the surface, to construct an interactive experience with a sense of hierarchy.

In terms of style, the page emphasizes the integration of simplicity, modernity and a sense of history: The color scheme adopts low-saturation aqueous tones (such as light gray, off-white, and lake blue), echoing the geographical image of the Grand Canal.

The font selection takes into account both readability and a sense of history.

The accompanying pictures adopt a collage style and Chinese meticulous painting style, emphasizing the relevant elements of the canal's dissemination and depicting the scenes related to the canal.

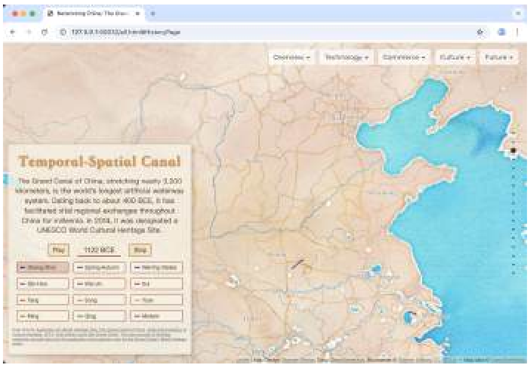
1.2.2 Page Structure and Interactive Modules

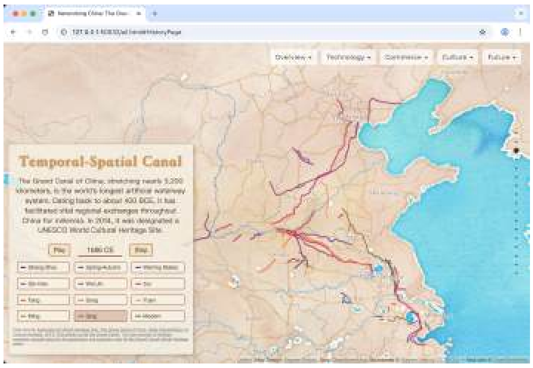
Introduction and Theme Presentation

The homepage directly introduces the theme concept that "The Grand Canal is a historical network", and the "Start Exploring" button is prominently placed below the introduction as a clear interactive entry point.

Temporal-Spatial Canal Visualisation

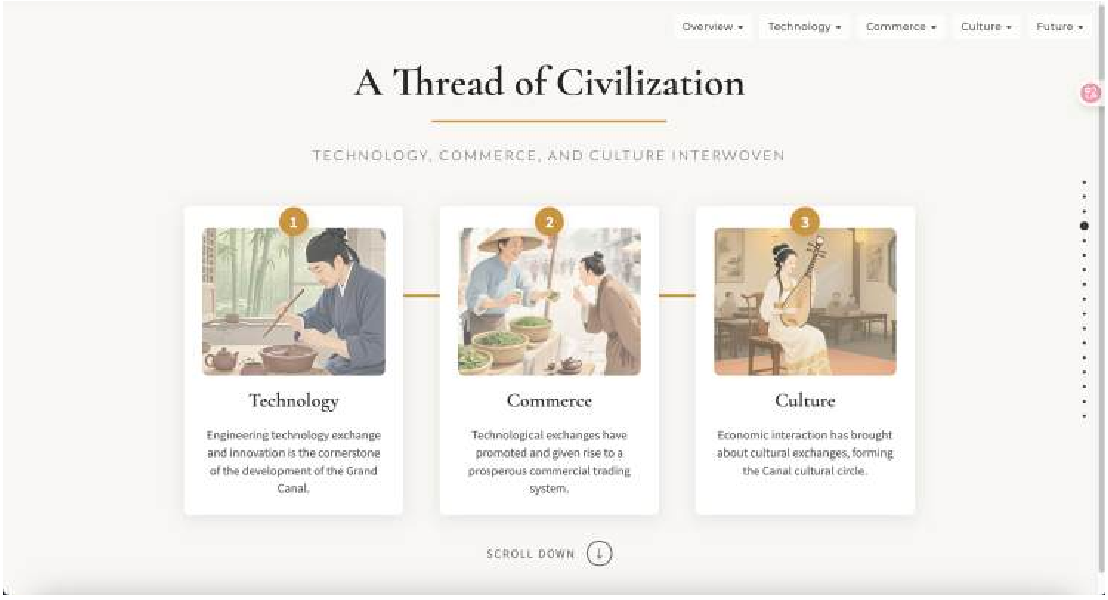
This interactive visualisation presents the Grand Canal as a dynamic spatial network spanning multiple dynasties. Using modern city names (e.g., Kaifeng for Bianjing) for clarity, it maps the Canal’s expansion based on official geo-data from the the Grand Canal HGIS Platform. The base map uses Stamen Watercolor map tiles to evoke traditional Chinese landscape aesthetics, with soft blue shading highlighting basin integration with five major river systems—Yangtze, Yellow, Huai, Hai, and Qiantang. A vertical sidebar lists major canal segments, allowing users to highlight routes aligned with the Canal’s north-south orientation. A time-based animation enables exploration of dynastic construction from the Shang-Zhou era to the modern age(1600BCE—256BCE), with colour-coded filtering and a “Play” function to animate historical growth. Interactive features such as clickable segments and dynasty-linked pop-ups enhance temporal-spatial understanding. Built with Leaflet.js and D3.js, the map ensures smooth, responsive interaction grounded in verified historical sources.





Network deconstruction

To enhance the intuitive perception of the "progressive relationship of online communication", we have designed three narrative dynamic scenes of selling tea with a purple clay teapot and playing the zither, each representing a different form of communication. These scenes are presented in sequence through a timeline animation, creating a flowing logic of civilization from "tools to transactions, and from transactions to spirit". Then enter the network page and click on the detailed introduction of each type of dissemination.

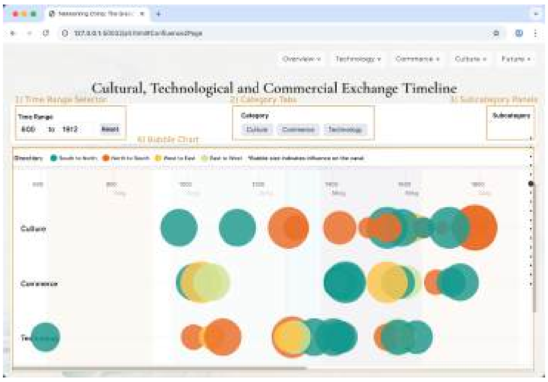


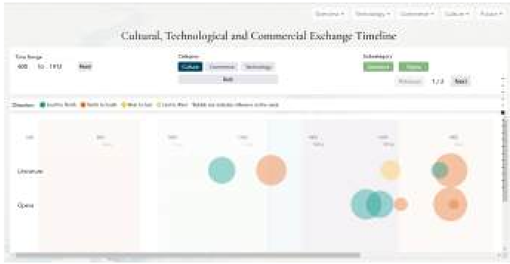
Cultural, Technological and Commercial Exchange Timeline

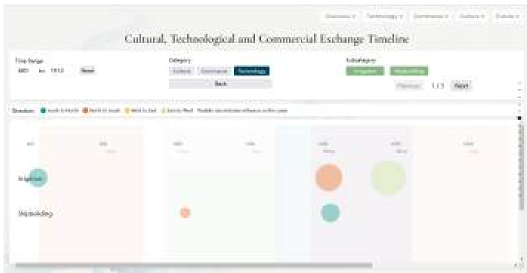
This section presents an interactive timeline visualizing key cultural, technological, and commercial exchanges along the Grand Canal between 600 and 1912 CE. Grounded in scholarly research (Li, 2023, Gao 2012, Zheng 2019, M 2025, Jianmei, Mengqi & Jia 2022), it showcases significant events or innovations closely linked to the Canal's role as a conduit for communication and integration across imperial China. This timeline provides users with a synthetic yet data-rich view of how the Grand Canal facilitated multidimensional exchanges over centuries. It transforms abstract academic research into a form that is visually intuitive and narratively compelling. By integrating temporal, spatial, directional, and categorical data, the timeline offers a holistic and comparative lens through which users can grasp the Grand Canal's role as a dynamic infrastructure of connection—not just for goods, but for ideas, practices, and innovations.

These bubble charts not only offer a high-level overview but also form the foundation for subsequent subvisualizations in each thematic module (Culture, Commerce, Technology). Users can filter events by time range, main category, and subcategory, with each bubble representing an event whose size reflects the Canal's influence and color indicates the direction of exchange. The layout allows intuitive comparison across dynasties and domains, with dynasty bands and year markers providing temporal context.

The interface is designed for clarity and interactivity, featuring dynamic category toggling, pagination for subcategories, and responsive tooltips on event selection. The minimalist style, enhanced by Tailwind CSS, ensures readability, while React state management enables real-time updates and transitions as users explore the dataset.

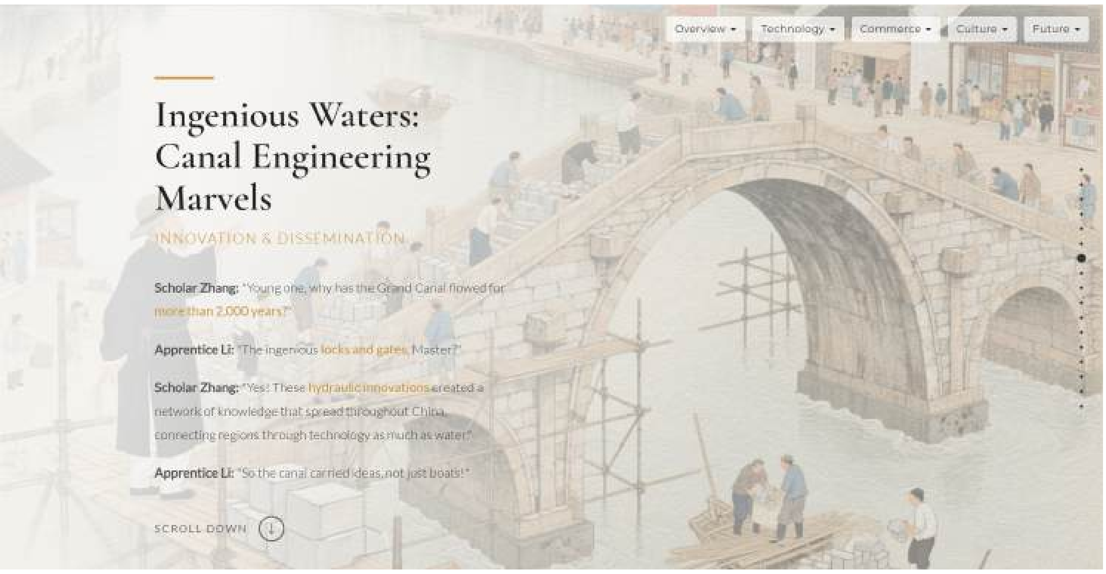






Network introduction

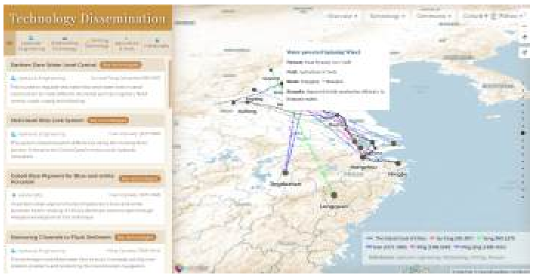
These three introductory web pages package serious historical and cultural knowledge into heuristic educational content through dialogues between different identities such as scholars and students, merchants and travelers, poets and artists, and arouse readers' interest in exploration.

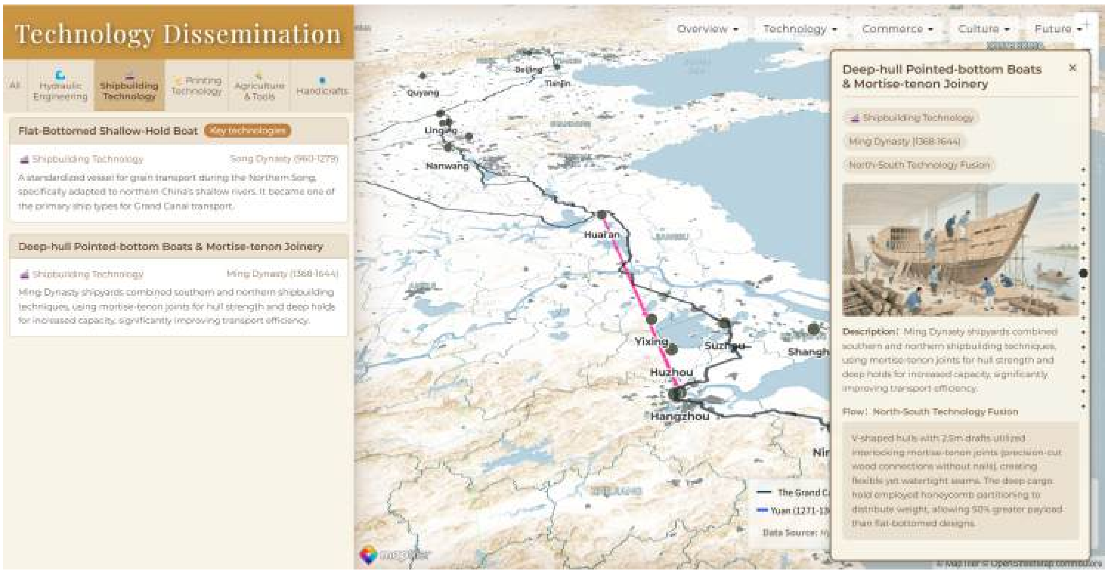


Then further abstract and structure each type of network and answer the following questions:
Where does the connection occur? How do technologies/commodities/cultures spread? Which nodes have a higher degree of "intermediary" or "influence"? How does the canal constitute a "structured civilization transmission system"?

Technology dissemination

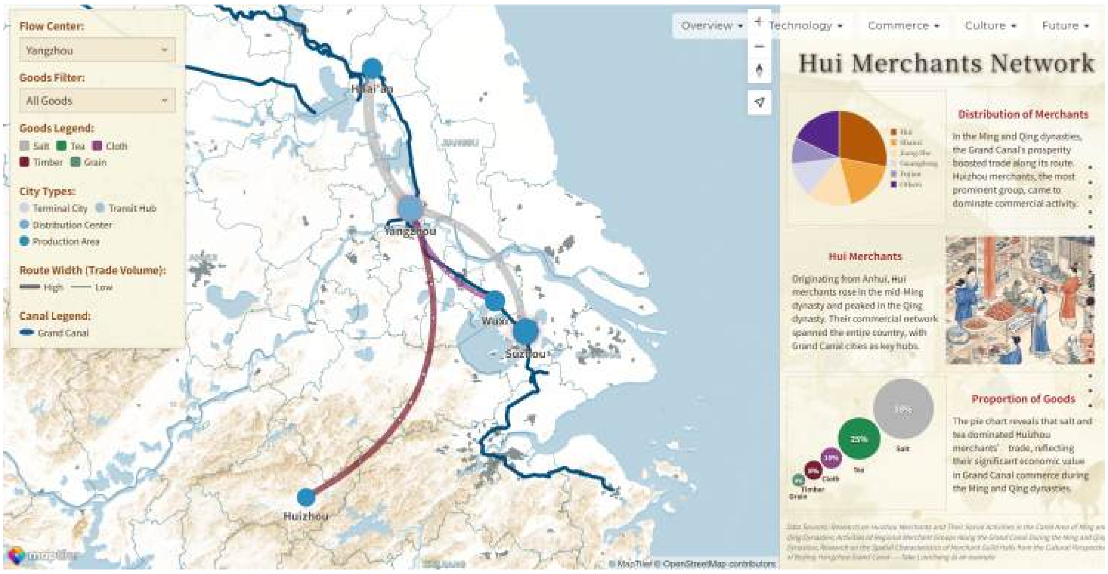
This section showcases the dissemination networks of various engineering technologies from different dynasties in the areas along the canal. The left panel builds a multi-level classification menu. All technologies are divided into five major categories, with representative technologies marked. The technology dissemination paths of different dynasties are distinguished by different colors. Five major categories, 18 technologies, supporting folding and floating prompts. Each technology has a uniformly styled technical illustration to enhance visual recognition. Click on a certain technology and a technology card will appear. In conjunction with the map and technical cards, the perspective is focused on the propagation path and highlighted, with arrows indicating the propagation direction. The technical cards also label each type of technology with classification, time and direction tags, and explain the key points of engineering technology. The content is based on data from master's theses (Mei, 2025; Yao, 2023), official government websites, historical archives, and documentary sources.





Commerce Network

To demonstrate the commercial distribution pathways of Huizhou merchants along the Grand Canal during the Ming and Qing dynasties, I constructed an interactive geospatial flow network map. By organizing trade data from academic research(Zheng,2020), I visualized the trade flows of five major commodity categories. The technical implementation uses the MapTiler API to build the geographical base map, SVG to draw trade routes, and particle animations achieved through Bézier curves and requestAnimationFrame to create flow effects. City markers employ radial gradients and size encoding to distinguish functional types. In terms of interface design, the right-side information panel displays the distribution of various merchant groups through pie charts, highlighting the dominant position of Huizhou merchants, while bubble charts show the five major commodity types. The left-side control panel features a trade center selection menu, supporting perspective switching among eight cities, and commodity filters allow individual viewing of specific goods circulation networks. Clicking on city markers triggers information cards that display city types and commercial roles. The map is linked with the control panel - when selecting trade centers, the view automatically focuses and highlights related pathways. Particle flow directions indicate commodity flow direction and density, while line thickness encodes trade volume intensity, creating an intuitive commercial network exploration experience. This map clearly presents the pivotal role and expansion patterns of Huizhou merchants within the Grand Canal commercial system.



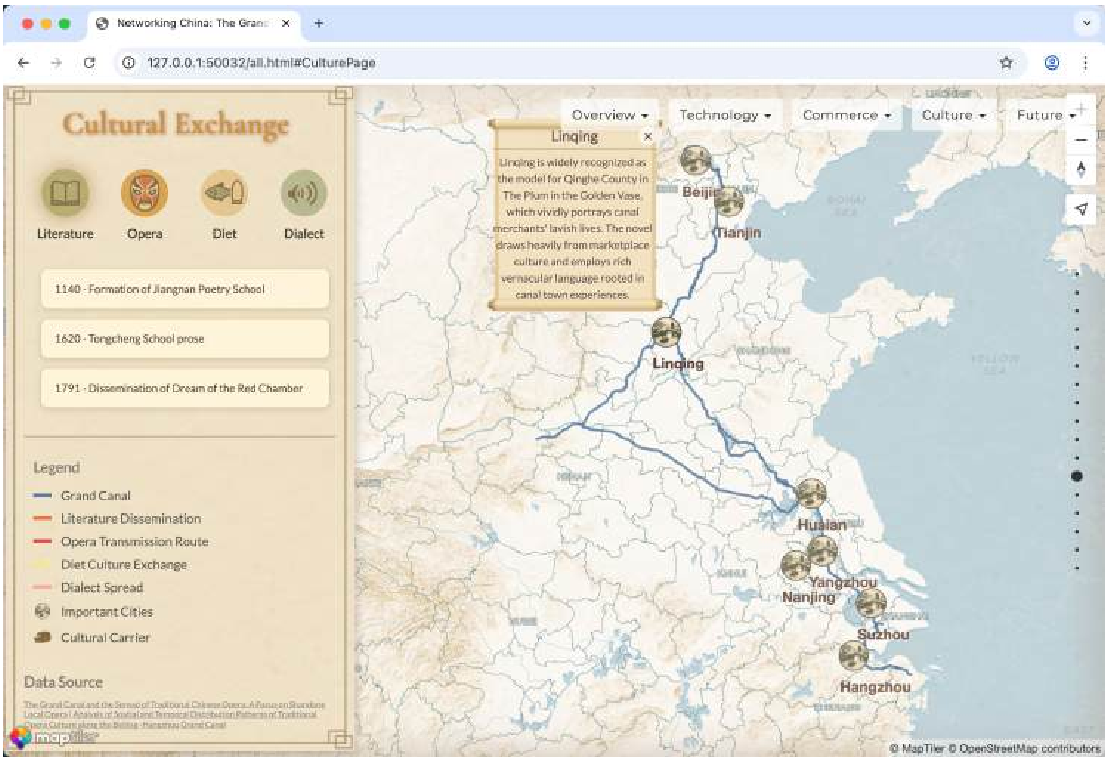
Cultural Exchange

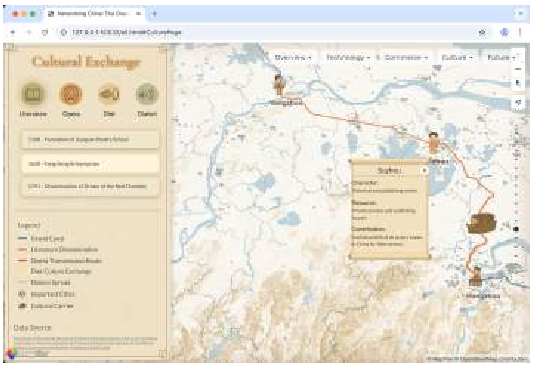
The Grand Canal facilitated the movement of intangible cultural assets across dynasties and regions, as illustrated by this interactive cultural exchange map, which concentrates on four key domains: Literature, Opera, Diet, and Dialect. Three representative events are displayed in each domain through a combination of spatial visualisations and temporal markers, including annotated city popups, cultural nodes, and thematic routes.

The initial view highlights literature, showcasing key dissemination cities such as Suzhou and Hangzhou. Each cultural domain has its own icon and colour-coded transmission route: red for opera, yellow for diet, and brown for dialect, with cities connected by historically plausible paths. Events are marked by year and description (e.g., “1791 - Dissemination of Dream of the Red Chamber”), and clicking them activates corresponding animated routes and zoomed map views that shift pitch, bearing, and centre position to emphasise the movement direction.

Each city node is enhanced with stylised illustrations relevant to the theme, such as traditional performers for opera or symbolic dishes for food culture. For instance, in the tea diffusion visualisation under Diet, different tea types serve as regional icons: Hangzhou with Longjing, Suzhou with Biluochun, Yangzhou with Kui Longzhu, and Beijing with Jasmine Tea—symbolising how the Canal merged production with cultural geography.

Deeper contextualisation of the function of each city is provided by popups. These encompass the city's historical significance, fundamental resources, and character. Users are capable of comprehending both the micro-level local significance and the macro-level movements through this format. The map utilises MapTiler as its cartographic foundation and incorporates custom icons, animated lines, and narrative views through interactive JS frameworks. Each perspective (event) employs a customised camera angle to enhance spatial narration, such as tilting southward to track the southbound spread of official-style cuisine.



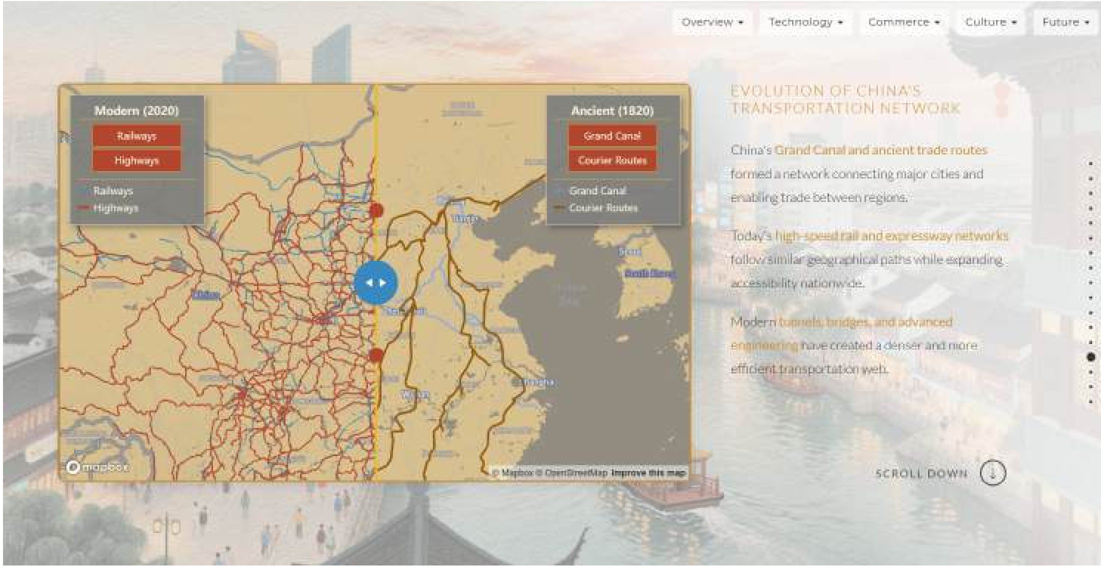


Traffic Transfer

This section serves as a transitional chapter from historical review to modern applications, demonstrating the historical evolution of the Grand Canal's freight functions through comparative display, laying the groundwork for subsequent cultural tourism transformation content. I used Chart.js to construct a dual-axis line chart, comparing freight volume changes across three transportation modes - canal, railway, and highway - through different Y-axis scales, employing smooth curves and gradient fills to enhance visual impact. The left text area uses layered animation effects to progressively reveal key transformation nodes, highlighting the glorious past of 70% freight share, the impact moment of railway's 4-fold capacity, and the stark reality of today's 60-fold gap. The right chart area clearly presents the fundamental transformation of transportation patterns from 1500-2023 through three different colored curves, establishing a data foundation for introducing the theme of "Grand Canal Renaissance." The bottom scroll prompt guides users to continue exploring. The overall design connects past and present, naturally transitioning from the reality of freight decline to a new chapter of cultural value rediscovery and tourism function development.

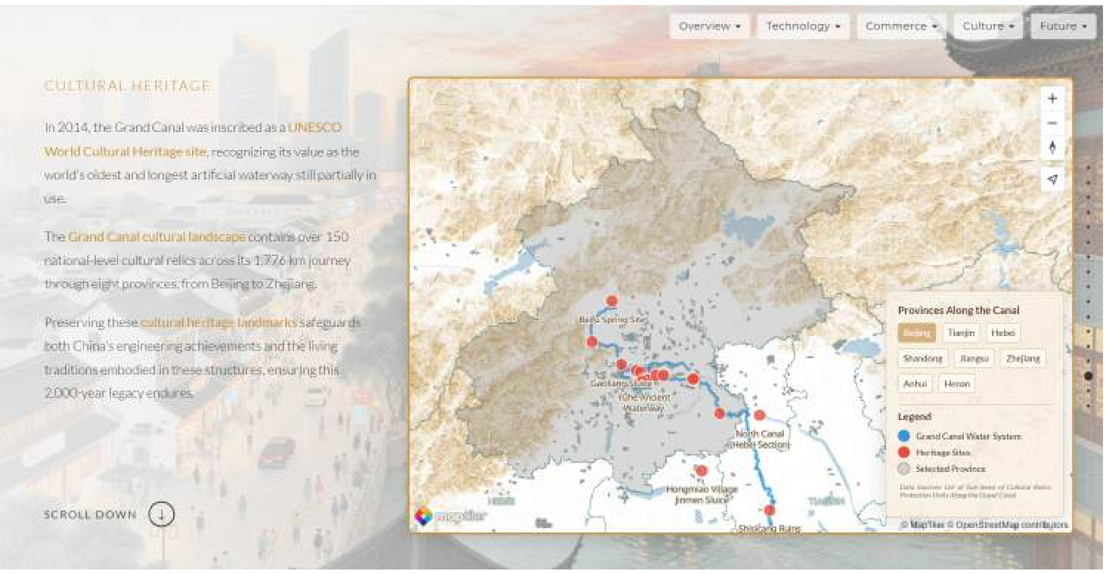
Comparison

This section demonstrates the evolution of China's transportation networks through ancient-modern comparison, revealing how modern transportation continues the geographical logic of ancient pathways. Technically, it uses Mapbox GL JS to build a dual-map comparison system, implementing left-right sliding transitions through the mapbox-gl-compare plugin. The webpage displays the 2020 modern transportation network on the left side and the 1820 ancient transportation system on the right, with a central slider supporting user-adjustable comparison ratios. Both sides feature legends providing layer toggle functionality, allowing users to independently switch different transportation types for precise comparison. The right text area employs fade-in animations to highlight key comparison points: ancient networks connecting major cities, modern networks following similar geographical paths, and engineering breakthroughs overcoming terrain limitations to create denser networks. The overall design, through intuitive visual comparison, showcases the transportation revolution from waterways and post roads to railways and highways, providing historical depth perspective for the subsequent repositioning of the Grand Canal's cultural value.



Protection

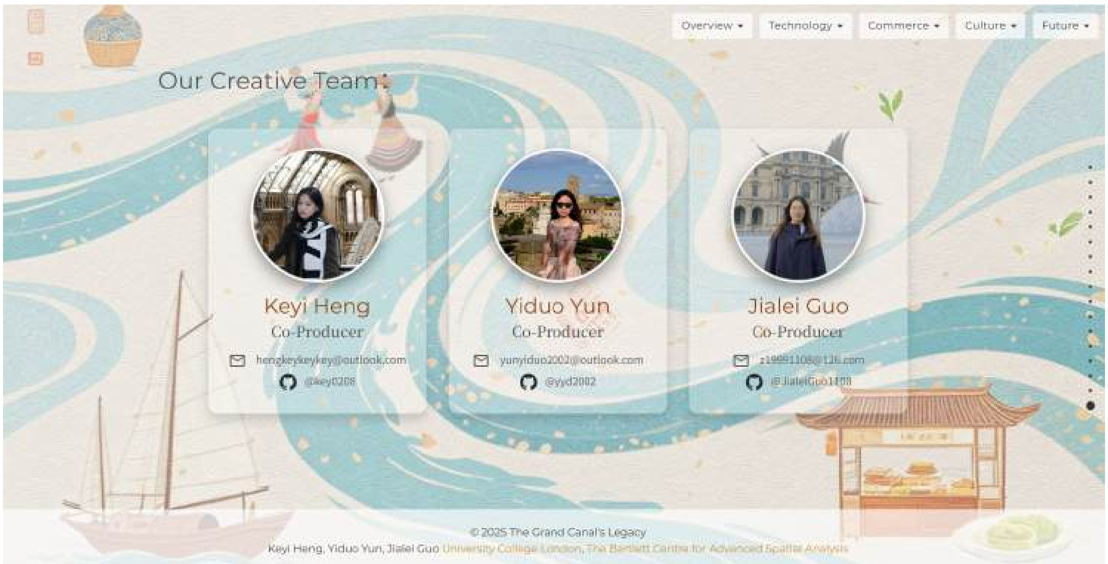
This section demonstrates the successful transformation of the Grand Canal from traditional freight functions to cultural tourism functions, presenting the distribution of cultural heritage and tourism resources across eight provinces and cities along the route through an interactive map. Using MapTiler SDK to build an interactive map, it achieves regional focus functionality through dynamic provincial boundary loading and map perspective switching. The right control panel provides selection functionality for eight provinces along the route - when users click on a province, the map automatically flies to position and highlights the provincial boundaries while loading corresponding cultural heritage sites. Blue lines mark the Grand Canal waterway system, red dots display cultural heritage sites, and gray semi-transparent areas highlight the selected province. Clicking on heritage sites triggers detailed information cards containing names and descriptions. This webpage showcases the tourism value and cultural inheritance significance brought by the Grand Canal as a World Cultural Heritage site, completing its magnificent transformation from economic artery to cultural corridor.



Epilogue

This module, as the closing part of the web page, adopts a rhythmic creative short essay as the conclusion, echoing the beginning, elevating the theme, and guiding users to shift their understanding of "the Grand Canal as a historical network system" towards in-depth thinking on the concept of "connection". It is not only a summary of the presented content, but also an open response to the future perspective and modern significance.

|



1.2.3 Data Sources

Data Category	Source Name	URL
Canal Paths & Historical Context	2013 Application for World Heritage	http://www.gchgis.com/CanalMap/index.html

Data Category	Source Name	URL
Important City Nodes	UNESCO World Heritage List	https://whc.unesco.org/en/list/1443/
Overall Canal Data	World Cultural Heritage Organization	https://www.wochmoc.org.cn/contents/32/1826.html
Hydraulic Engineering	Academic Research Paper	https://xueshu.baidu.com/usercenter/paper/show?paperid=111d06s0uf7n0gb0ag0g04t0sc343725&site=xu
Shipbuilding Technology	Academic Research Paper	https://xueshu.baidu.com/usercenter/paper/show?paperid=1h270p901t400am0vv0n0jh013559118&site=xu
Printing Technology	Video Documentary	https://www.youtube.com/watch?v=H9jXqFtrmpM
Cultural Relics	Canal Museum Digital Collection	https://canalmuseum.net/diancang.html
Cultural Studies	Culture-Ecology-Tourism Research	Academic Paper (Li et al., 2022)
Canal Culture	Grand Canal Cultural Heritage	http://www.cngrandcanal.cn/?history/967.html
Literature Studies	Dream of the Red Chamber Research	Academic Paper (Dong, 2021)
Opera Culture	Chinese Opera Spread Study	Academic Paper (Zhou, 2019)
Opera Artifacts	Brick Carving Collection	https://www.chnmus.net/ch/collection/appraise/details.h id=512155583365899677
Folk Songs	Intangible Cultural Heritage Study	https://www.nature.com/articles/s40494-023-00911-w
Culinary Culture	Grand Canal Food Culture Research	Academic Paper (Feng, 2023)
Dialect Studies	Wu Dialects Research	Academic Paper (Chao, 1967)
Huizhou Merchants Research	Academic Paper (Ming-Qing Period)	https://www.zywhyj.cn/Uploads/PdfFile/2020-07-24/5f1a
Regional Merchant Groups	Yangzhou University Research	https://dyhyjy.yzu.edu.cn/info/1021/3028.htm

Data Category	Source Name	URL
Merchant Guild Halls	Spatial Characteristics Study	https://d.wanfangdata.com.cn/thesis/D03034666
Statistical Data	National Bureau of Statistics	https://www.stats.gov.cn/
Railway Data	China Railway Corporation	https://www.china-railway.com.cn/
Transportation Data	Ministry of Transport	https://www.mot.gov.cn/
Geographic Data	ArcGIS World Map	worldmap.maps.arcgis.com
Cultural Heritage Sites	Wikipedia - Grand Canal Heritage List	https://zh.wikipedia.org/wiki/大运河文物保护单位子项列表

1.2.4 Visualization Methods and Tools

Technology Stack	Application Module / Scenario	Brief Description
Mapbox + D3.js	Technology diffusion, cultural routes	Base map with interactive path highlighting, direction arrows, and card linkage
MapTiler + SVG + Animation	Commercial network (Huizhou merchants)	Flow path visualization with particle animation for trade direction and density
TimelineJS / React Component	Multi-theme timeline (culture/tech/trade)	Time filtering, category toggling, bubble size shows influence
Leaflet.js + GeoJSON	Temporal-Spatial Canal Map	Dynamic historical canal expansion by dynasty
Mapbox GL JS + Compare	Ancient vs. Modern Transport Comparison	Side-by-side sliding maps with layer toggling
Chart.js	Freight function transformation	Dual-axis line chart comparing canal, rail, and road freight volumes
MapTiler + Popup Cards	Cultural heritage & tourism map	Clickable provinces show sites with auto-fly and detail cards
SVG/Canvas Animation	Network structures, dynamic flows	Path rendering, directional arrows, node interaction
HTML/CSS/JavaScript	Global UI & interaction control	Module switching, pop-ups, thematic styling and transitions

2 Technical Challenges and Solutions

2.1 Card-Map Highlighting and Focusing Linkage

In developing the "Technology Dissemination Network" module, the linkage mechanism between cards and the map is crucial for enhancing the user experience. When a user clicks on a technology card, the map needs to immediately:

Highlight the dissemination path of the corresponding technology and de-emphasize all other paths. Automatically focus the map view on the dissemination route area of that technology.

2.1.1 How to accurately locate map path data from card information?

Since map path data is stored in the attribute fields of GeoJSON, precise highlighting requires matching both "dissemination content" and "dynasty" fields.

Solution: Use the map.querySourceFeatures() method to filter the GeoJSON data source based on conditions, and combine it with setPaintProperty to dynamically modify the style of the target path for highlighting.

In [ ]:

```
const matchingFilter = [
  'all',
  ['==', ['get', 'Disseminate content'], techName],
  ['==', ['get', 'Dynasty'], period]
];

const features = map.querySourceFeatures('technology-route', {
  filter: matchingFilter
});
```

2.1.2 How to automatically focus the map view on the path?

The path is a GeoJSON LineString, and its coordinates are a two-dimensional array of coordinates. It's necessary to calculate its bounding box and execute fitBounds(), while also preventing the technology card from obscuring it. Solution: Calculate the minimum and maximum longitude and latitude values of the coordinates to dynamically set the bounds of the bounding box. Use asymmetric padding to offset the map focus area to the left, reserving space for the cards on the right.

In [ ]:

```
map.fitBounds(bounds, {
  padding: {
    top: 150,
```

```
    bottom: 150,
    left: 55,
    right: 315  // Space reserved for right-side cards
  },
  duration: 2000,
  maxZoom: 8
});
```

2.1.3 How to establish data flow connections between the map and cards?

It's essential to implement a bidirectional event communication mechanism: from clicking a technology card → map focusing, and from clicking a map path → activating the corresponding card. Solution: Use CustomEvent + window.dispatchEvent() to build an event bus system: Triggered by Technology Browser:

In [ ]:
```
const techSelectedEvent = new CustomEvent('techSelected', {
  detail: { techName: 'xxxx', period: 'xxxxx' }
});
window.dispatchEvent(techSelectedEvent);
```

Map listens for events and links map path highlighting:

In [ ]:
```
window.addEventListener('techSelected', function(e) {
  const techDetail = e.detail;
  highlightTechRoute(techDetail); // Call map handling function
});
```

At the same time, when clicking a path on the map, also trigger an event to notify the right-side technology panel in reverse:

In [ ]:
```
const event = new CustomEvent('mapTechSelected', {
  detail: { techName: techName, techId: findTechIdByName(techName) }
});
window.dispatchEvent(event);
```

2.2 Coordinate Synchronization Issues in Dynamic Trade Route Rendering

2.2.1 Dynamic Coordinate Synchronization for SVG Paths

When developing the trade flow path visualization functionality, I encountered a key technical challenge: when users zoom or pan the map, all trade paths drawn through SVG would experience position displacement or disappear. This occurs because MapTiler's coordinate projection system changes with view changes, while SVG paths use a fixed pixel coordinate system, creating a synchronization mismatch between the two. Initially, I attempted a static drawing approach, rendering all paths at once when the map loaded, but quickly discovered this method produced serious display errors during user interaction. The core of the problem lay in the coordinate transformation within the calculateFlowPath function:

In [ ]:

```
function calculateFlowPath(fromCity, toCity, index, total) {
    const fromPoint = map.project([fromCity.lng, fromCity.lat]);
    const toPoint = map.project([toCity.lng, toCity.lat]);
    const dx = toPoint.x - fromPoint.x;
    const dy = toPoint.y - fromPoint.y;
    const dist = Math.sqrt(dx * dx + dy * dy);

    const curveFactor = 0.25 + (index % 3) * 0.1;
    const curveDirection = index % 2 === 0 ? 1 : -1;
    const midX = (fromPoint.x + toPoint.x) / 2;
    const midY = (fromPoint.y + toPoint.y) / 2;
    const controlX = midX + (-dy * curveFactor * curveDirection);
    const controlY = midY + (dx * curveFactor * curveDirection);

    return {
        path: `M${fromPoint.x},${fromPoint.y} Q${controlX},${controlY} ${toPoint
        points: { from: fromPoint, to: toPoint, ctrl: { x: controlX, y: controlY
        distance: dist
    };
}
```

Here, the pixel coordinates returned by the map.project() method change with map view changes, causing SVG paths to become invalid. My solution was to establish a complete dynamic redrawing mechanism. By listening to the map's move events, whenever the map view changes, the system automatically recalculates coordinate points for all trade routes, uses Bézier curve algorithms to generate new path data, and then updates the d attribute of SVG elements. To avoid calculation conflicts, the current animation frame is canceled before recalculation begins, and the animation loop is restarted after calculations are complete. While this approach adds some computational overhead, it completely resolves coordinate synchronization issues, ensuring users can see correct trade routes under any view.

In [ ]:

```
map.on('move', () => {
   if (!centerMarker) return;

   // Step 1: Pause current animation to avoid calculation conflicts
   if (animationFrameId) cancelAnimationFrame(animationFrameId);

   // Step 2: Batch recalculate all trade paths
   flowLines.forEach((flow, index) => {
       const src = cityData.find(c => c.name === flow.source);
       const tgt = cityData.find(c => c.name === flow.target);
       if (!src || !tgt) return;

       // Recalculate Bézier curve path
       const pathData = calculateFlowPath(src, tgt, index, flowLines.length);
       flow.element.setAttribute('d', pathData.path);

       // Step 3: Synchronously update particle animation path reference points
       if (particleGroups && particleGroups[index]) {
           particleGroups[index].pathPoints = pathData.points;
```

```
        }
    });

    // Step 4: Restart animation loop
    animateParticles();
});
```

2.2.2 Performance Optimization Challenges in Particle Animation System

When implementing dynamic particle effects for trade flows, I encountered serious performance issues. The initial design created a fixed number of animated particles for each trade route, but when simultaneously displaying complex networks from multiple trade centers, the page could have hundreds of particles animating at once, causing browser frame rates to plummet dramatically and creating a poor user experience. Traditional DOM animation methods were completely unsuitable for this scenario, requiring the search for more efficient solutions. After performance testing and optimization experiments, I first implemented an intelligent particle quantity control algorithm. The core code of this algorithm is as follows:

In [ ]:

```
function createParticles(pathData, color, particleCount, flowGroup) {
    // Dynamically calculate particle count based on path distance and trade volu
    const particleCount = Math.min(
        Math.max(3, Math.floor(tradeVolume * 2.5)),
        Math.floor(pathData.distance / 15)
    );

    const particleContainer = document.createElementNS('http://www.w3.org/2000/sv
    const particlesArray = [];

    for (let i = 0; i < particleCount; i++) {
        const particle = document.createElementNS('http://www.w3.org/2000/svg', '
        const size = 2 + Math.random() * 2;
        particle.setAttribute('r', size / 2);
        particle.setAttribute('fill', color);
        particle.setAttribute('class', 'particle');

        // Assign different initial positions and speeds to each particle
        const startPos = Math.random();
        const baseSpeed = (0.0005 + Math.random() * 0.0015);

        particlesArray.push({
            element: particle,
            pos: startPos,
            speed: baseSpeed * defaultAnimationSpeed,
            baseSpeed: baseSpeed,
            size: size
        });

        particleContainer.appendChild(particle);
    }

    flowGroup.appendChild(particleContainer);
```

```
    return { group: particleContainer, particles: particlesArray };
}
```

The ingenious aspect of this algorithm lies in establishing three constraints: a minimum of 3 particles to guarantee visual effects, a maximum not exceeding the path distance divided by 15 to avoid excessive density, while considering trade volume factors for dynamic adjustment. This ensures both visual effects and avoids unnecessary performance overhead. Next came the optimization of the animation loop. I used requestAnimationFrame to replace traditional timers and implemented efficient particle position calculations:

In [ ]:

```
function animateParticles() {
    particleGroups.forEach(({ particles, pathPoints }) => {
        particles.forEach(p => {
            // Update particle position on Bézier curve
            p.pos += p.speed;
            if (p.pos > 1) p.pos = 0;

            // Calculate particle's current coordinates on Bézier curve
            const currentPoint = getPointOnQuadraticCurve(
                pathPoints.from,
                pathPoints.ctrl,
                pathPoints.to,
                p.pos
            );

            // Directly update DOM attributes, avoiding complex CSS animations
            p.element.setAttribute('cx', currentPoint.x);
            p.element.setAttribute('cy', currentPoint.y);
        });
    });

    // Use requestAnimationFrame to ensure synchronization with browser refresh r
    animationFrameId = requestAnimationFrame(animateParticles);
}

function getPointOnQuadraticCurve(p0, p1, p2, t) {
    const omt = 1 - t;
    const x = omt * omt * p0.x + 2 * omt * t * p1.x + t * t * p2.x;
    const y = omt * omt * p0.y + 2 * omt * t * p1.y + t * t * p2.y;
    return { x, y };
}
```

The key to this optimization approach lies in using mathematical formulas to directly calculate particle positions on Bézier curves, avoiding complex CSS animations or easing functions, which significantly reduces computational overhead. Meanwhile, requestAnimationFrame ensures animation synchronization with the browser refresh rate, preventing unnecessary redraws.

2.3 Technical Challenges in Animating Cultural Routes and Displaying City Information

During the development of the Cultural Exchange webpage, several technical challenges emerged in balancing visual richness with interactive functionality. Key issues involved animating custom routes smoothly and presenting dense city-level information within a stylized interface.

2.3.1 Complex Path Animation Synchronization

Animating a boat marker along a complex GeoJSON path was difficult because standard GeoJSON LineString or MultiLineString formats don't support point-by-point time control or smoothing. The animation often appeared jerky or the boat deviated from the path due to uneven segment lengths or projection distortion. The route points were first extracted from the GeoJSON and total distance was calculated. A custom linear interpolation function (getPointAtDistance) was used to compute the precise location of the boat at any point in time. This ensured the boat moved smoothly and proportionally along the route, regardless of segment length.

The route points were first extracted from the GeoJSON and total distance was calculated. A custom linear interpolation function (getPointAtDistance) was used to compute the precise location of the boat at any point in time. This ensured the boat moved smoothly and proportionally along the route, regardless of segment length.

This approach allowed precise and fluid movement of the animated boat marker, significantly improving visual realism and user experience.

In [ ]:
```
function getPointAtDistance(targetDistance) {
  let currentDistance = 0;
  for (let i = 0; i < routePoints.length - 1; i++) {
    const segmentDistance = calculateDistance(routePoints[i], routePoints[i + 1]
    if (currentDistance + segmentDistance >= targetDistance) {
      const remaining = targetDistance - currentDistance;
      const ratio = remaining / segmentDistance;
      return [
        routePoints[i][0] + (routePoints[i + 1][0] - routePoints[i][0]) * ratio,
        routePoints[i][1] + (routePoints[i + 1][1] - routePoints[i][1]) * ratio
      ];
    }
    currentDistance += segmentDistance;
  }
  return routePoints[routePoints.length - 1];
}
```

In [ ]:
```
function loadRoute(type, file, color, iconImage, viewParams) {
    d3.json(file).then(data => {
        map.addSource(`route-${type}`, { type: 'geojson', data: data });
        map.addLayer({ id: `line-${type}`, type: 'line', source: `route-${type}`

        // Perspective setting
        const center = computeCenterOfRoute(data.features[0].geometry.coordinate
        map.flyTo({
            center: center,
```

```
            zoom: viewParams.zoom,
            pitch: viewParams.pitch,
            bearing: viewParams.bearing,
            duration: 1500
        });

        // Start the route animation
        extractRoutePoints(data);
        createBall(iconImage);
        startAnimation();
    });
}
```

2.3.2 Multi-field Content Display in Styled Popups

Each city node contains multiple text fields (e.g., character, resource, contribution) that needed to be shown within a stylized scroll-like popup. The original implementation reused a generic popup for both icon and text-triggered events, but this failed to properly display longer content or apply distinct visual formatting.
Initially, both types shared the same popup design, but visual and structural limitations led to separating them for better control and user experience.
The solution was to create two differentiated popup types: one for overview (icons) and one for detail (textboxes).
The textbox version uses a `<div>` with a custom background image (scroll texture) and explicitly sets `style.overflowY = 'auto'` and max-height to ensure longer content scrolls within a fixed area. This maintains visual integrity while enhancing readability.

In [ ]:
```
popup.className = 'city-info-popup textbox-source';
popup.style.maxWidth = '320px';
popup.style.maxHeight = '380px';
popup.style.overflowY = 'auto';  // Enable scroll for long text
popup.style.backgroundImage = 'url(gjlimages/scroll9.png)';
popup.style.backgroundSize = '100% 100%';
```

2.4 Technical Challenge in Interactive Page Design

During the final phase of project development, we encountered a core technical challenge: how to deliver a consistent and intuitive user interaction experience across different types of pages.

The project features two main types of pages, each with specific interactive requirements:

2.4.1 Full-Screen Map Pages

These include interactive map components powered by MapTiler.

- Core issue: The scrolling behavior for section navigation conflicts with the zoom and pan interactions on the map.
- Examples: Technology Dissemination Map, Commercial Network Map, Cultural Exchange Map.
- Solution: Click-based paging mechanism

In [ ]:
```
HTML (Map Page: Click to Navigate Sections)
<!-- Map page: click arrows to change sections -->
<div class="section" id="TechPage">
    <iframe src="hky_tech.html" title="Technology Dissemination"></iframe>

    <!-- Scroll indicator: click only, no scroll -->
    <div class="scroll-indicator">
        <div class="scroll-arrow"> </div>
    </div>
</div>
```

In [ ]:
```
JavaScript
// Click handler: manually trigger fullPage.js paging
document.querySelector('.scroll-indicator').addEventListener('click', function(e
    e.preventDefault();
    e.stopPropagation(); // Prevent conflict with map zoom
    if (window.parent && window.parent.$.fn && window.parent.$.fn.fullpage) {
        window.parent.$.fn.fullpage.moveSectionDown();
    }
});
```

2.4.2 Standard Content Pages

These include static visual content, timelines, and small interactive maps.

- Interaction requirement: Natural scroll-based navigation
- Examples: Intro pages, transitional pages, timeline visualizations
- Solution: Scroll-triggered paging mechanism + Click-based pagingmechanism

In [ ]:
```
HTML (Content Page: Scroll to Navigate)
<!-- Standard content page: scroll triggers paging -->
<div class="section" id="Tourism">
    <iframe src="yun_tour.html" title="Tourism Industry"></iframe>

    <!-- Scroll trigger zone: overlays bottom area -->
    <div style="position: absolute; bottom: 0; left: 0; width: 100%;
                height: 180px; z-index: 2;
                background: linear-gradient(to bottom, transparent, rgba(0,0,0,0
    </div>

    <!-- Visual indicator: scroll hint only, no click -->
    <div class="scroll-indicator">
        <div class="scroll-arrow"> </div>
    </div>
</div>
```

References

Yao, K. (2023) A study on shipbuilding along the Huaiyang section of the Grand Canal during the Song and Yuan dynasties. [Master’s thesis] Jiangsu University of Science and Technology.
Mei (2025) conducted a comprehensive study on the identification and conservation of hydraulic engineering heritage along the Beijing section of the Grand Canal.
Jianmei, L.I., Mengqi, S. & Jia, Q.I., 2022. Coupling Coordination Development of Culture-Ecology-Tourism in Cities along the Grand Canal Cultural Belt, Economic geography, årg. 42, nr. 10, s. 201–207.
Yan, H., 2021. The making of the Grand Canal in China: beyond knowledge and power, International Journal of Heritage Studies. Routledge, årg. 27, nr. 6, s. 584–600.
Jianmei, L.I., Mengqi, S. & Jia, Q.I., 2022. Coupling Coordination Development of Culture-Ecology-Tourism in Cities along the Grand Canal Cultural Belt, Economic geography, årg. 42, nr. 10, s. 201–207.
Zhang, S., Zhang, X. & Liu, J., 2024. Spatial distribution and pedigree age of intangible cultural heritage along the Grand Canal of China, Heritage Science. Nature Publishing Group, årg. 12, nr. 1, s. 1–20.
Zhou, A., 2019. The Grand Canal and the Spread of Chinese Opera: A Study Focused on Local Operas in Shandong. Drama (Journal of the Central Academy of Drama), 2019(04), pp. 149–159. DOI: 10.13917/j.cnki.drama.2019.04.014.
Gao, S., 2012. A Study on the Waterway Spread of Chinese Opera: Taking the Grand Canal as an Example. Mass
Hangzhou Municipal Government, 2020. Implementation Plan for the Protection, Inheritance, and Utilization of the Grand Canal Culture in Hangzhou. Official Planning Document, Hangzhou, China.