



Progressive Web Apps with Angular

渐进式网页应用程序与Angular

December, 2016



Angular exists to make
building applications
with the web easy.

Angular 使建立网页应
用程序变得更容易.



Angular Everywhere






Agenda

1.App Shell Generation App Shell生成

2.Service Worker 服务工

3.Push 推送

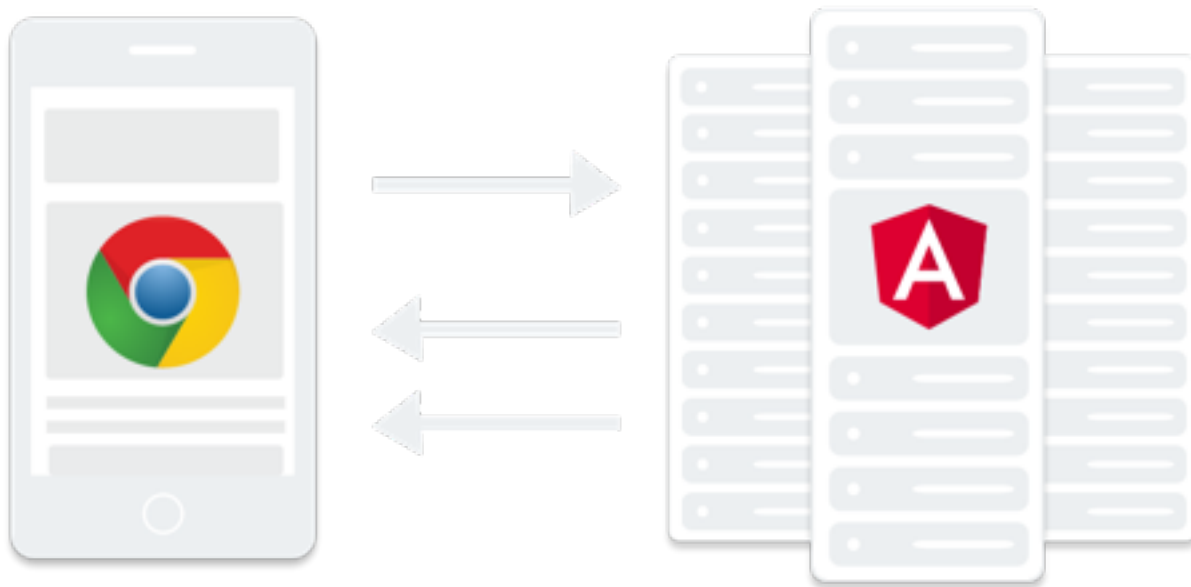
4.Live Coding 现场编程



App Shell Generation

App Shell 生成







AppShellModule

```
import {AppShellModule} from '@angular/app-shell';
import {NgModule} from '@angular/core';
import {CommonModule} from '@angular/common';

import {RootComponent} from './root';
import {WeatherModule} from './weather';

@NgModule({
  bootstrap: [RootComponent],
  declarations: [
    RootComponent,
  ],
  imports: [
    AppShellModule,
    CommonModule,
    WeatherModule,
  ],
})
export class AppModule {}
```




Directives

```
<header class="header">
  <h1 class="header__title">Weather PWA</h1>
  <button (click)="refresh()" id="butRefresh" class="headerButton" aria-label="Refresh"></button>
  <button (click)="showPicker()" id="butAdd" class="headerButton" aria-label="Add"></button>
</header>

<main class="main" *shellNoRender>
  <weather-card *ngFor="let city of cities" [city]="city"></weather-card>
</main>

<main class="main" *shellRender>
  <h2>Loading...</h2>
</main>

<div class="dialog-container" [ngClass]="viewState">
  <city-picker [cities]="weatherData.allCities" (save)="addCity($event)" (cancel)="onCancel($event)"></city-
picker>
</div>
```



Directives

```
<header class="header">
  <h1 class="header__title">Weather PWA</h1>
  <button (click)="refresh()" id="butRefresh" class="headerButton" aria-label="Refresh"></button>
  <button (click)="showPicker()" id="butAdd" class="headerButton" aria-label="Add"></button>
</header>

<main class="main" *shellNoRender>
  <weather-card *ngFor="let city of cities" [city]="city"></weather-card>
</main>

<main class="main" *shellRender>
  <h2>Loading...</h2>
</main>

<div class="dialog-container" [ngClass]="viewState">
  <city-picker [cities]="weatherData.allCities" (save)="addCity($event)" (cancel)="onCancel($event)"></city-picker>
</div>
```

Loading...

Dom Tree & Critical path CSS

```
<!DOCTYPE html>
<html>
  <head>_</head>
  <body>_
    <ngw-root _ngghost-2e37-1>
      <header _ngcontent-2e37-1 class="header">
        _</header>
        <!--template bindings={}-->
        <!--template bindings={}-->
      <main _ngcontent-2e37-1 class="main">
        <h2 _ngcontent-2e37-1>Loading...</h2>
      </main>
      <div _ngcontent-2e37-1 class="dialog.container">
        <city-picker _ngcontent-2e37-1 _ngghost-2e37-2>
          <div _ngcontent-2e37-2 class="dialog">_</div>
        </city-picker>
      </div>
    </ngw-root>
  <universal-script>
    <script>_</script>
  </universal-script>
</body>
</html>
```



Service Worker 服务工



Registration 注册

```
<script defer>
  if (navigator['serviceWorker']) {
    navigator.serviceWorker.register('/worker-basic.js');
  }
</script>
```



ngsw-manifest.json

```
{
  "routing": {
    "routes": {
      "/": {
        "prefix": false
      }
    },
    "index": "/index.html"
  }
}
```



gulpAddStaticFiles

```
gulp.task('task:service-worker', () => gulp
  .src('ngsw-manifest.json')
  .pipe(gulpAddStaticFiles(gulp.src([
    'dist/**/*.js'
  ]), {manifestKey: 'static'}))
  .pipe(gulp.dest('dist'))
);
```




File Caches

Application

- Manifest
- Service Workers
- Clear storage

Storage

- Local Storage
- Session Storage
- IndexedDB
- Web SQL
- Cookies

Cache

- Cache Storage
 - ngsw:active - http://localhost:8000
 - manifest:6a753c4e2bfa72b062d74571407fcf7965ca637d:static - http://localhost:8000
 - ngsw:staged - http://localhost:8000
- Application Cache

Frames

- top



Coming Soon: Dynamic Content Caching 动态内容缓存

Cache URLs which are encountered at runtime

缓存运行时遇到的网络连接



Push 推送

ServiceWorkerModule 服务工模块

```
@NgModule({  
  imports:      [  
    BrowserModule,  
    RouterModule.forRoot([]),  
    AppShellModule.runtime(),  
    CommonModule,  
    ServiceWorkerModule,  
  ],  
  bootstrap:    [AppComponent],  
})  
  
export class AppModule {}
```



registerForPush 推送注册

```
NgServiceWorker.registerForPush( )
```



Push Observable 推送可见

```
NgServiceWorker.push: Observable<message>
```



Push Messages 推送信息

```
export class AppComponent {  
  push;  
  msgs;  
  constructor(sw: NgServiceWorker) {  
    if (sw) {  
      sw.registerForPush().subscribe(reg => {  
        this.push = reg.toJSON();  
      });  
      this.msgs = sw.push.scan((msgs, value) => {  
        msgs.push(value);  
        return msgs;  
      }, []);  
    }  
  }  
}
```



Live Coding 现场编程



See More

- Full Angular PWA Sample

<https://github.com/alxhub/ng2-weather-pwa>

Thank You!

