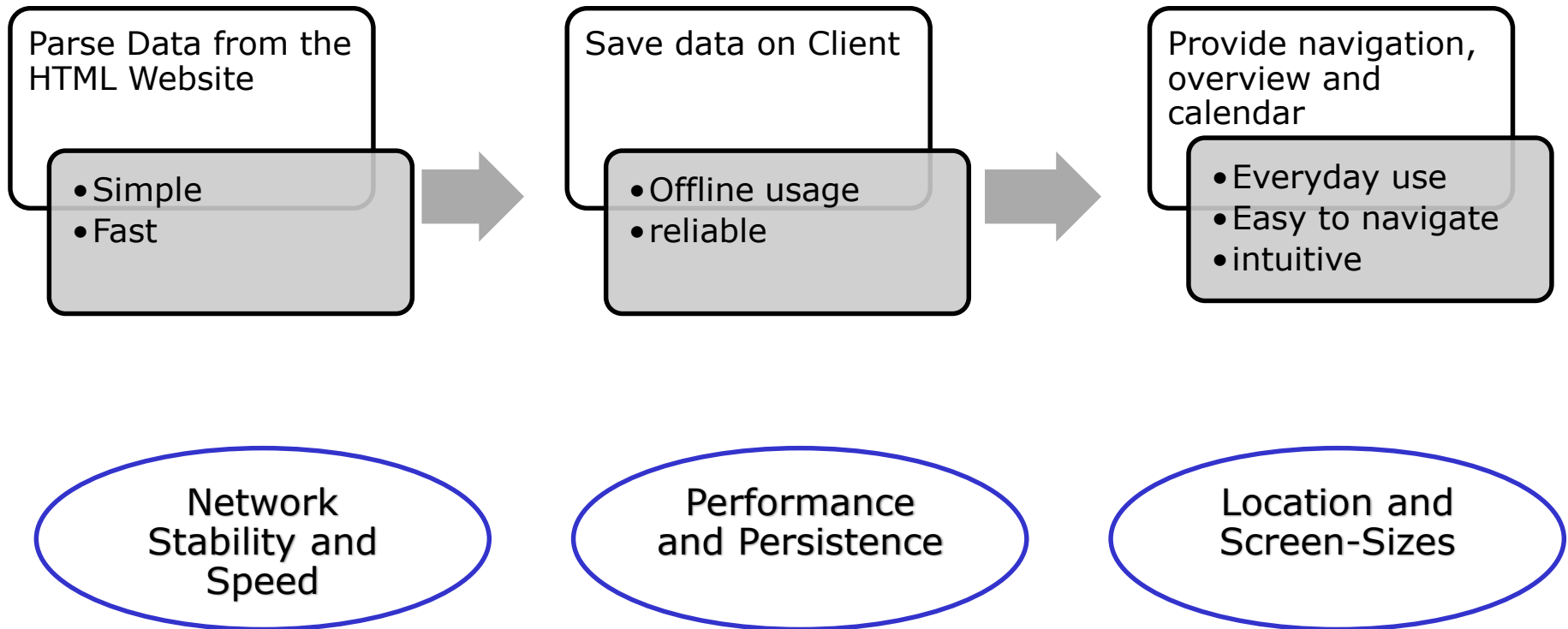# Application Development for Mobile and Ubiquitous Computing

Seminar Task
Final Presentation

GroupNo. 1
Team: Benjamin Kurczyk, Hendrik Thüs
"HVS-App"

TECHNISCHE UNIVERSITÄT DRESDEN

Parse Data from the HTML Website		Save data on Client		Provide navigation, overview and calendar
• Simple • Fast	→	• Offline usage • reliable	→	• Everyday use • Easy to navigate • intuitive

Network Stability and Speed

Performance and Persistence

Location and Screen-Sizes

➢ Present the Data from the HVS in a readable way

➢ Provide fast and easy update of game results

➢ Provide navigation to the game locations

➢ Show League Overviews and League Standings

- **Efficient HTML Parsing**
  - No API for easy communication
  - Need to communicate loading time with user

- **Adaption to different screens**
  - More than 19.000 different devices with Android[1]
  - Need to adapt to different screen sizes and resolutions

- **Using location and time context**
  - GPS sensor, Google maps and playground location
  - Integration of user calendar

1: http://www.telekom-presse.at/software_apps/android-fragmentierung_waechst_um_60_prozent_auf_rund_19-000_verschiedene_modelle.id.31496.htm

- **The Parser is running in reasonable time**
  - The app is context aware and warns on low network speed
  - Bulk Database operations increase speed a lot
  - (Exp: TelephonyManager.*NETWORK_TYPE)*

- **The App is adapting to different screen sizes**
  - Usage of „wrap:parent" and other XML attrbiutes in layouts
  - Only very tiny screen are problematic (text folded too often)

- **Locationbased functions check for suitable providers**
  - User gets warned on bad location providers ()
  - Routing is user choice
  - (Exp: Criteria.*ACCURACY_FINE)*

```java
private static boolean isConnectionFast(int type, int subType) {
    if (type == ConnectivityManager.TYPE_WIFI) {
        return true;
    } else if (type == ConnectivityManager.TYPE_MOBILE) {
        switch (subType) {
        case TelephonyManager.NETWORK_TYPE_1xRTT:
            return false; // ~ 50-100 kbps
        case TelephonyManager.NETWORK_TYPE_CDMA:
            return false; // ~ 14-64 kbps
        case TelephonyManager.NETWORK_TYPE_EDGE:
            return false; // ~ 50-100 kbps
        case TelephonyManager.NETWORK_TYPE_EVDO_0:
            return true; // ~ 400-1000 kbps
        case TelephonyManager.NETWORK_TYPE_EVDO_A:
            return true; // ~ 600-1400 kbps
        case TelephonyManager.NETWORK_TYPE_GPRS:
            return false; // ~ 100 kbps
        case TelephonyManager.NETWORK_TYPE_HSDPA:
            return true; // ~ 2-14 Mbps
        case TelephonyManager.NETWORK_TYPE_HSPA:
            return true; // ~ 700-1700 kbps
        case TelephonyManager.NETWORK_TYPE_HSUPA:
            return true; // ~ 1-23 Mbps
        case TelephonyManager.NETWORK_TYPE_UMTS:
            return true; // ~ 400-7000 kbps
            /*
             * Above APT level 7, make sure to set android:targetSdkVersion
```

- The app is dependent on the design of the parsed website

- Androids HTTPRequest handling needs a lot of fine tuning and checks

- In terms of performance, every action on the DB counts!

- Tabular Data can be organized well with Fragments and TabViews

- Increasing amounts of data will push towards a server-client solution in the future