ISSN:0975-9646

A Recent Approach to Organise Structured Data in Mobile Environment

R.Jegadeesan [1], Dr.N.Sankar Ram[2], R.Janakiraman[3]

Ph.D Research Scholar [1], Supervisor[2], Assistant Professor[3]
Anna University-Chennai (India)[1],
Anna University-Chennai (India)[2],
The New Royal college of Engineering and Technology Chennai (India)[3]

Abstract:- A Novel Approach To Organize Structured Data In Mobile Devices propounds an ISAMD (*Improved Synchronization Algorithm based on Secured Message Digest*) in order to facilitate data synchronization between a server-side database and a mobile database. The ISAMD algorithm generates a Message Digest Table (MDT) at the server database and the mobile database using Secure Message Digest. The Message Digest Table holds the encrypted form of data stored by the user. These Message Digest Tables are compared row-wise in order to select the rows needed for synchronization. If and when any two rows of these MDT are different, the synchronization process progresses according to the synchronization policy. The ISAMD algorithm does not use techniques that are dependent on specific database vendors; neither does it use triggers, stored procedures or timestamps. The ISAMD uses only the standard SQL functions for the synchronization. The ISAMD algorithm can be used in any combinations of server-side database and mobile database because of its independency of database vendor. The encrypted Data stored in the server database can be decrypted and retrieved upon user's request.

Keywords: wireless networks, computer networks and others.

INTRODUCTION:

Recent advances in mobile technology and computing fields have led to the emergence of a new computing environment and a variety of small sized mobile devices such as PDAs (personal digital assistants), smart mobile phones, HPCs (handheld PCs) and Pocket PCs have been popularized. As various network technologies are increasingly being associated with such mobile devices, the processing of enterprise business information and solution can be available using handheld mobile devices. As a result, business models that rely on mobile technologies are appeared.

Mobile devices do not have much computing power and rely on batteries. Additionally, constant access to network is difficult due to narrow bandwidth. Therefore, it is not easy to process a large size of stored data and maintain a continuous connection with the server-side database. For these reasons, mobile devices have mobile databases in order to achieve stable data processing.

Mobile devices download replications of limited data from a connected server-side database using a synchronization device that has a stable wire communication function. Mobile devices process various tasks using the data downloaded in an off-line state.

The work on the network disconnected condition is a crucial point for mobility support. In a disconnected environment, there are inevitable inconsistencies between the server-side database and mobile database. Synchronization techniques can solve the data inconsistencies and guarantee the integrity of data. Consequently, synchronization is an essential subject in mobile device computing environments.

Commercial DBMS vendors offer various solutions to data synchronization in a mobile environment. However, these solutions are not independent of the server-side database because they use database dependent information such as metadata or use specific functions of server-side database such as trigger and time stamp. In other words, the mobile database vender should be equivalent to the server-side database vendor.

The solution of operating a separate synchronization server in the middle tier is independent of the server-side database but dedicated to the mobile database. That is, the synchronization solution and the mobile database should be the identical vendor product. Additionally, when a client programmer develops mobile applications that are embedded in mobile devices, the developer should uses a particular library that is provided by the vender of mobile database or modify existing mobile applications for synchronization process. Because of these restrictions, the extensibility, adaptability and flexibility of mobile business systems are markedly decrease. This problem must be solved in order to build efficient mobile business systems because upcoming mobile environments will have heterogeneous characteristics in which diverse mobile devices, mobile databases, and RDBMS exist.

This paper suggests Improved Synchronization Algorithms based on Message Digest (ISAMD) in order to resolve the problems mentioned above. ISAMD resolves synchronization problems using only standard SQL queries as certified by the ISO (International Organization for Standardization). This is followed by a possible synchronization of any data combination regardless of the kind of server-side database or mobile database. The ISAMD therefore would provide extensibility, adaptability and flexibility. The ISAMD algorithm generates a Message Digest Table (MDT) at the server database and the mobile database using Secure Message Digest. The Message Digest Table holds the encrypted form of data stored by the user. These Message Digest Tables are compared row-wise in order to select the rows needed for synchronization. If the value of message digest regarding identical rows is different for both sides, it means the duplicated rows have been changed and synchronization is necessary using ISAMD. The Message Digest Table holds the encrypted form of data stored by the user. The encrypted Data stored in the server database can be decrypted and retrieved upon user's request. Message digest is used to detect falsification of data transferred mainly via security protocols.

In this procedure, because a large volume of data is compressed into a small volume, we can simplicity the detection of data inconsistencies and minimize wasted storage space. Message digest functions work fast even with limited resources, so that they reduce the burden placed on mobile devices that have small computing power.

II. RELATED WORKS:

In the literature, some schemes have been proposed for Mobile computing has become a reality thanks to the convergence of two technologies: the appearance of powerful portable computers and the development of fast reliable networks. In the mobile wireless computing environment of the future massive number of low powered computer machines will query databases over the wireless communication channels. In the author presents a novel synchronization mechanism for multi-field programmable gate array (multi FPGA) simulation accelerators with time-multiplexed interconnection are presented. The proposed event-based synchronization mechanism reduces synchronization time among multiple FPGAs. A mobile e-business client application may intentionally operate in disconnected mode to reduce the communication cost and the power consumption of mobile devices. We use "data hoarding" to allow business transactions to be processed on the mobile client despite of disconnection, which is implemented by the materialized view. We split the synchronization of a mobile transaction with the server database into 2 steps: intermediate and final synchronization. The intermediate synchronization uses the high cost wireless medium while the final synchronization generally uses the low cost communication medium such as a wired LAN. In this paper suggests SAMD (Synchronization Algorithms based on Secured Message Digest) in order to resolve the problems mentioned above. SAMD resolves 938 synchronization problems using only standard SQL queries as certified by the ISO (International Organization for Standardization). This is followed by a possible synchronization of any data combination regardless of the kind of database of server side or mobile database. In conclusion, the SAMD is effective solution for mobile database synchronization in ubiquitous environment. In an OFDM baseband receiver for DVB-T/H is presented. The receiver contains four synchronizations, OFDM symbol synchronization, a carrier synchronization, a sampling clock synchronization and a scattered pilots synchronization.

TABLE I: COMPARITIVE RESULTS

Author and Year	Solution	Strength	Weakness
Barbara.D, in 1999	Adaptive Clock Synchronization on scheme	Independence of vendors	Slower indexing
My-Sun Choi, Young-Guk Kim in 2001	Event based synchronization scheme	Good indexing	Poor extensibility
Sang-ouk Kim, Se-Bong Oh in 2002	Time synchronization scheme	Better extensibility	Synchronization is very slow
EPFL,U.Grenoble,INRIANancy in 2004	Low complexity	Better synchronization	Security Is less
Joshua Savill in 2008	FPGA scheme	Best security	Slow Time consumption
Mi-Young Choi 2010	Synchronization improvements	Best security	Difficult access

This paper proposes several novel designs to reduce the synchronization latency and hardware complexity. The carrier and clock synchronization loops are fully digitalized schemes. The scattered pilots synchronization adopts a two stages scheme to reduce the detection latency. In addition, the pre-filling scheme reduces the latency of channel estimation.

III. PROBLEM AND THE PROBLEM SOLVING TECHNIQUES

A.EXISTING SYSTEM

1. Commercial DBMS venders offer various solutions to data synchronization in a mobile environment.
2. However, these solutions are not independent of the server-side database because they use database dependent information such as metadata or use specific functions of server-side database such as trigger and time stamp.
3. Because of these restrictions, the extensibility, adaptability and flexibility of mobile business systems are markedly decrease.

EXISTING ALGORITHMS:

1. Adaptive Clock synchronization algorithm
2. Event based synchronization algorithm
3. Time synchronization algorithm
4. Low Complexity algorithm

DRAWBACKS OF EXISTING SYSTEM:

1. Slower indexing
2. Poor extensibility, adaptability and flexibility
3. Synchronization is very slow
4. Dependent on database vendors
5. Security is less
6. Higher Cost
7. Slow Time consumption.

B.PROPOSED SYSTEM:

1. ISAMD resolves synchronization problems using only standard SQL queries. This is followed by a possible synchronization of any data combination regardless of the kind of server-side database or mobile database.
2. The ISAMD makes the images at the table of the server-side database and the mobile database using a Secured Message Digest algorithm; then the images, and the Secured Message Digest values, are saved in the Secured Message Digest tables on both sides.
3. The ISAMD algorithm compares two images in order to select the rows needed for synchronization.
4. Secured MessageDigest is used to detect falsification of data transferred mainly security protocols.

ADVANTAGES OF ISAMD:

1. Independence of vendors.
2. Synchronization using only standard SQL statements.
3. Disallows schematic modification of data table of the server-side database.
4. Disallows adding restrictions in implementing applications.

IV.PERFORMANCE ANALYSIS

The goal of the proposal is to provide security to the data with the attributes of independence of venders, synchronization using only standard SQL statements, avoiding schematic modification of data table of the server-side database, adding restrictions implementing applications are avoided. Precisely, to build a more efficient processing of business information in mobile devices using ISAMD.

MODULES:

1. Implementation of mobile
2. Implementation of server
3. Synchronization of server

Implementation of mobile: To Design a mobile (client) for synchronization processing. It contain user business information. This database that can be connected to by a mobile computing over a mobile network.

Implementation of server: To Design a server model for synchronization processing. Server database maintains all of the data required for business applications.

Synchronization of server: Synchronize mobile client database to server database to achieve higher security using ISAMD (Synchronization Algorithm based on Secured Message Digest).

V.SYSTEM IMPLEMENTATION

ISAMD Algorithm:

The Proposed system has a server-side database and the mobile database where the ISAMD synchronization algorithm is applied. Both databases have data table (DSDT, MCDT) and a Secured Message Digest table (DSMDT, MCMDT). The data
table contains the business data, and the Secured Message Digest table stores the Secured Message Digest value from the data table. The Secured Message Digest table consists of a PK column of data table, Secured Message Digest value (MDV) column, flag (F) column and mobile device ID (Mid) column.

The flag column signals an inconsistency that has occurred in the corresponding column; therefore, the flag column is used to identify a row that requires synchronization.

The mobile device ID is a unique number of the mobile device, so this column is used to identify a mobile device that requires synchronization. If a row's PK value is AI, this value is identical to the two Secured Message Digest values and there is no need for synchronization. However, if a row has a PK value of Cl, the value of MDV in MCMDT is different from the value of MDV in DSMDT and the MCMDT flag value is 1. Consequently, synchronization is necessary. The synchronization process is performed for each row to resolve all of the inconsistencies mentioned in Section II/B. For example, if there is an inconsistency in row Cl, synchronization takes place from the mobile database to the server-side database and DSDT's PK Cl row is replaced with the MCDT's Cl row.

The synchronization algorithm consists of Synchronizations 1, 2 and 3, involves synchronizing the data table and Secured Message Digest table. Therefore, the two are identical synchronization algorithms applied to different tables. Here, the Secured Message Digest values that are created with each row value of the data table, and the Secured Message Digest values of the Secured Message Digest table, are compared. If the values are identical, there has been no change in the data and synchronization is not necessary. If the values are different, it means that the data table value has been changed, in which case the Secured Message Digest table has to be updated with new Secured Message Digest values and the flag has to be set to 1. flag value is used to identify a row that needs synchronization. The server-side database has one DSMDT for every DSDT. Although the size of the MCMDT is smaller than that of the DSMDT, there is an MCMDT for every mobile device .It is very inefficient to perform Synchronization 2 for every row of the DSDT every time there that has a unique ID. Is a synchronization request from a mobile device.

Therefore when the mobile device requests synchronization, the mobile device ID value is sent to the server-side database and then the ISAMD algorithms select the row from DSMDT whose value of Mid column is the same as the mobile device ID value and Synchronization 2 is only applied to the selected rows. For example, a mobile device whose mobile device ID value is 'mdl' requests synchronization, the rows whose value of mid column is 'mdl' are selected and then only used in Synchronization 2. After ISAMD algorithms analyze the type of inconsistency using the flag values of both messages digest tables, primary key, which is used to identify the row. Therefore, Synchronization 3 is performed between two data tables for each inconsistent type upon completion of synchronization; the flag of the synchronized row is set to 0 in the Secured Message Digest table. Most mobile devices have limited resources, and the load on the device should be minimized during the synchronization process.

Accordingly, all Secured Message Digest tables are located in the server-side database to economize storage space of the mobile device, as shown in Fig., while there is the load caused by accessing the network in Synchronization 1 but the data size of MCDT is smaller than the server. Furthermore, the MCDT data necessary for Synchronization 1 is sent to the server-side database in a single transmission over a wired network using an SQL query capable of batch processing. After this point, there is no load on the mobile device, which reduces the load caused by network access in the Synchronization 1 stage.

SECURED MESSAGE DIGEST ALGORITHM:

1. Secured Message Digest is a cryptographic technique that consists of unidirectional hash function.
2. Map a message of random length to a fixed length.
3. The Secured Message Digest(h) is created by Hash function(H) which can be expressed as:
   a. h= H (M)
4. The Secured Message Digest is mainly used for security purpose (Encrypt the message).

Synchronization:

The synchronization algorithm consists:
Synchronization 1
Synchronization 2
Synchronization 3
Improvisation

Synchronizations 1 and 2: Synchronize the data table and Secured Message Digest table.

Synchronization 3: Synchronization 3 is performed between two data tables for each inconsistent type.

Improvisation: Improvisation is performed finally to reduce time complexity in the synchronized data tables.

ISAMD Synchronization framework:

The whole Synchronization framework consists of a server-sidedatabase, synchronization server (AnySyn) and multiple mobile devices with internal mobile databases.

SCHEMA OF ISAMD ALGORITHM

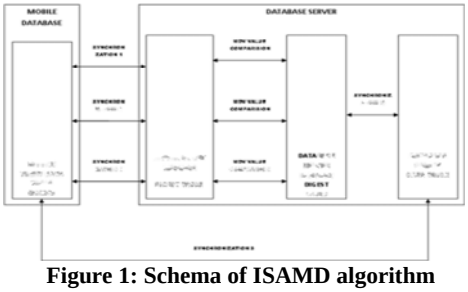

Figure 1: Schema of ISAMD algorithm

The server-side database maintains all of the data required for business, and the mobile database downloads copies of data the user needs from the server-side database. The synchronization server is located between the two databases to synchronize the data and manage additional information required for synchronization. Any synchronization server performs synchronization based on the ISAMD algorithm.The synchronization policy is established in Any Syn, and the load caused by accessing the server-side database is minimized by operating a connection pool. Every mobile device uses a separate toolkit to access the Any Synserver over a wired network to perform synchronization.

VI. SYSTEM DESIGN

A.CONCEPTUAL DESIGN

Software design is the first of the three technical activities designs, development and testing that is required to build software and verify its working. Design is an iterative process through which requirements are transformed into a blueprint for constructing the software. While designing, the software is logically partitioned into various modules that perform specific tasks required to obtain the output, figure 3 describes the data flow diagram for the ISAMD Algorithm.

DETAILED DESIGN:

Detailed design activity is concerned with the specification of algorithmic details, concrete data representation, interconnections among functions and data structures, and packaging the software product.

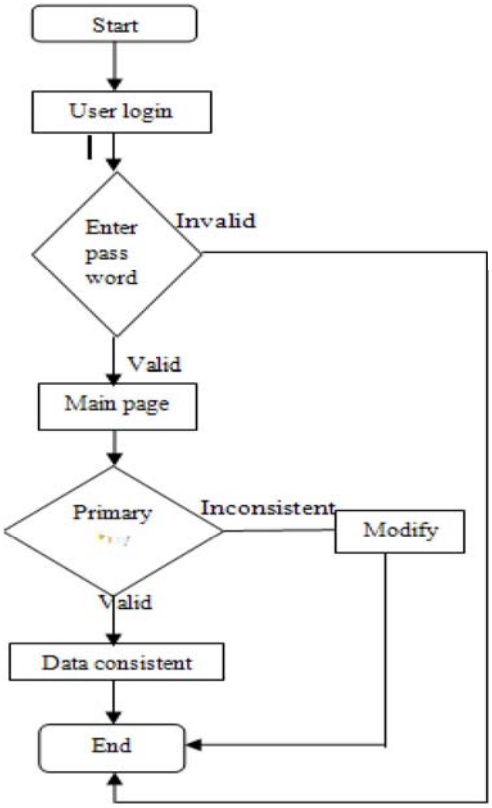

Figure 3: Data flow diagram for ISAMD

Detailed design is strongly influenced by the implementation language but it is not the same as implementation. Detailed design is more concerned with semantic issues and less concerned with syntactic issues than its implementation.

Detailed design starts with the architectural structure for which algorithmic details and concrete data representations are to be provided. There are many advantages in doing detailed design before going for implementation.

With detailed design, program control hierarchy and software execution procedure become clearer. Given the architectural and detailed design specifications, any programmer familiar with the implementation language will be able to do the implementation of the software. The detailed design of the ISAMD algorithm in the figure 4.

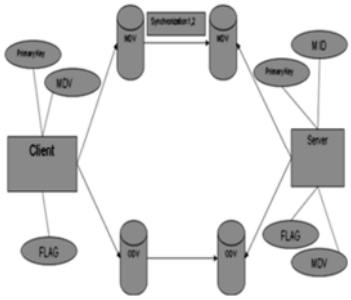

Figure 4: Detailed design

UML DIAGRAMS:

Every complex system is best approached through a small set of nearly independent views of a model; no single view is sufficient. Every model may be expressed at different levels of identity. The best models are connected to reality. The UML defines several graphical diagrams.

1. CLASS DIAGRAMS:

Class diagrams are the mainstay of object-oriented analysis and design. UML class diagrams show the classes of the system, their interrelationships (including inheritance, aggregation, and association), and the operations and attributes of the classes. Class diagrams are used for a wide variety of purposes, including both conceptual/domain modeling and detailed design modeling. Figure 5 explains the class diagrams.

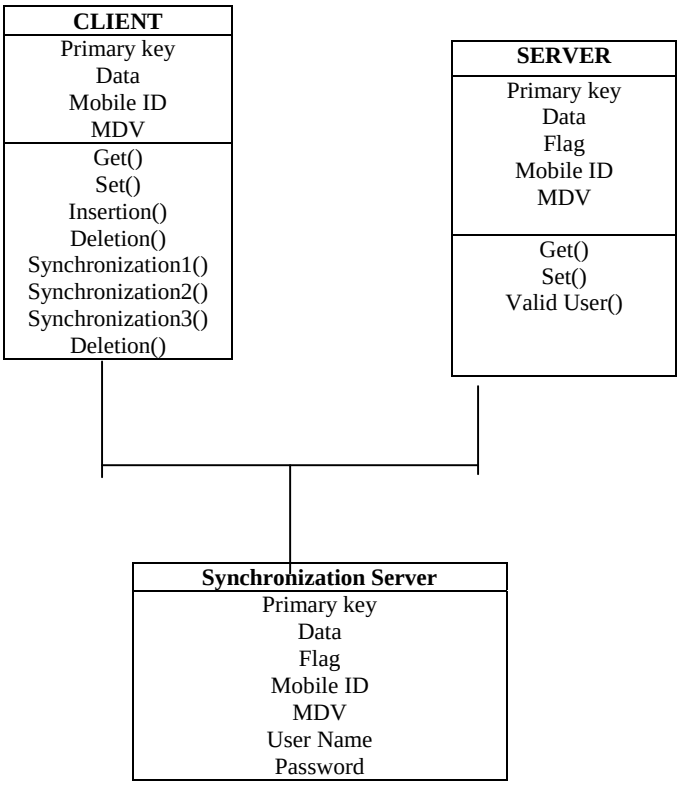

Figure 5: Class Diagram

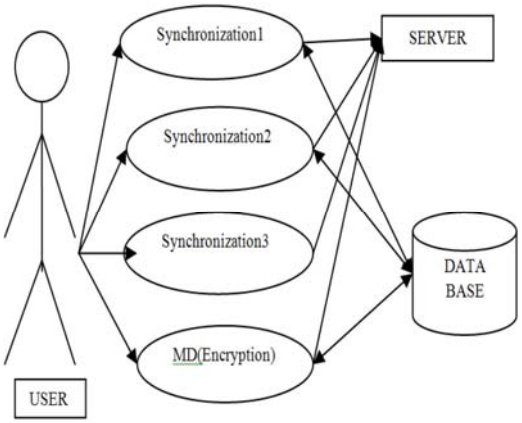

Figure 6: Use Case Diagram

2. USE CASE DIAGRAM:

UML use case diagrams overview the usage requirements for a system. They are useful for presentations to management and/or project stakeholders, but for actual development it provides significantly more value because they describe "the meat" of the actual requirements. Figure 6 explains the use case diagram.

VII. CONCLUSION

The Mobile databases are provided a better security of the data. The ISAMD algorithm merely uses the standard SQL queries of the relational database management system to perform the synchronization process. The ISAMD algorithm can be used with a combination of any mobile-side database and server-side databases. The ISAMD algorithm guarantees the extensibility, flexibility and adaptability. The project has been very interesting experience. Throughout the entire project we have gained extra knowledge about various advanced technologies, which we hope may be, useful for our career.

Future research directions:

1. Integration of Heterogeneous Applications to Mobile Devices.
2. Dynamic Database Storage improvements in Mobile Clients.
3. Simultaneous and Multiple Applications Processing.

REFERENCES:

[1] Sang-ouk Kim, Se-Bong Oh, Sung-Young Son, Jin-HoLee, "The framework for synchronization in embedded database environment.", Journal of Computing Science and Engineering, VoL20, Num.7,pp. 14-21, 2002.

[2] Tomasz Imielinski and B. R. Badrinath, "Mobile wireless computing: challenges in data management", Communications of the ACM, Volume 37 Issue 10, pp. 18-28, 1994.

[3] Barbara, D., "Mobile Computing and Databases – ASurvey", IEEE Transactions on Knowledge and Data Engineering, Vol. 11 No. 1, pp 108-117, 1999.

[4] My-Sun Choi, Young-Guk Kim, "Introduction of mobile database and research status", Journal of Database Research, Vol7, Num.3, pp. 3-16, 2001.

[5] Joshua Savill, "MobiLink Synchronization Profiles",A Whitepaper from Sybase in Anywhere. October 17th,2008.

[6] Thomas Fanghanel, Jonas S Karlsson, Cliff Leung, "DB2 Everyplace Database Release 8.1: Architecture and Key Features", Datenbank-Spektrum, pp. 1~15, 5/2003.

[7] A Database Synchronization Algorithm for Mobile Devices Mi-Young Choi, Eun-Ae Cho, Dae-Ha Park,Chang-Joo Moon, Doo-Kwon Baik IEEE Transactions Consumer Electronics, Vol. 56, No. 2, May 2010.

[8] John E. Canavan, "Fundamentals of Network Security", ARTECH HOUSE, INC., 2001, 61~62 Nov-2011

[9] Jonathan Knudsen, "WIRELESS JAVA: Developing with Java 2, Micro Edition", A press, 2001, pp. 155.

[10] "Mobile link Synchronization User's Guide", Synbase,Inc., 2004, pp. 37~68, pp. 392~402 March-2012

R.Jegadeesan has registered Ph.D (I&CE) Research in Anna University Chennai,India 2011 and his research interests are wireless networks and computer Networks. and he has received B.E. and M.E in computer science and Engineering in same University Chennai, India.

During his undergraduate and post graduate studies, he had spent time with so many research projects in many Universities in India, From 2000 to till date, also been contributed to many Universities for visiting faculty particularly Computer Networks and wireless networks related forum, currently working with the proposal of wireless networks in Anna University Chennai, India. in addition that, Fellowship of IEEE Member,Computer society of india(CSI), RedHAT Linux Instructor memberships for last three years. And Published many National and international level of Conferences, Journals, Seminars and Workshops.

Dr.N.SankarRam is the Professor, Computer Science and Engineering, working in a self financing Engineering college affiliated to Anna University, Chennai. He completed his under graduation and post graduation from Madurai Kamaraj University and Doctorate from Anna University Chennai. His research areas of expertise are Software Architecture and Computer Networks. He has 16 years of teaching experience and he has published many research papers in International/National journals and conferences. He is a approved supervisor under Anna University of Technology, Chennai and guiding Ph.D and M.E research scholars in his area. Being a Principal Investigator got a funded project to a tune of Rs. 9 Lakhs from AICTE under RPS scheme in the year 2011.

Mr.R.Janakiraman Presently working as an Assistant Professor in The New Royal College of Engineering and Technology,at chennai,india. In the department of CSE.he hasRegistered the Ph.D in Bharathiar university,coimbatore.

Received the P.G(M.Tech) in CSE from Vinayaga Mission university,salem(India).and M.Sc (CS&I.T) in Madurai Kamaraj University, Madurai-india. also has the Member of CSI, his research interest includes Computer Networks, Wireless Networks,Cloud Computing. And he Published many National and level of Conferences, Journals, Seminars and Workshops.