website www.jaypeebrothers.com, for detailed information on pediatric books, visit our website www.jaypeebrothers.com

Celebrating 50 YEARS
Passion, Quality and Innovation in Healthcare Publishing

Preventive Newborn Health

A global approach in the age of information

Editor-in-Chief
Balaji Govindaswami

Associate Editor
Robin Daughters Wu

Assistant Editor
Nisha Bedi Donley

Illustrative Editor
Albert H Alhatem

Section Editors
Glenn Henry DeSandre
Kamakshi Devarajan
Matthew J Garabedian
Priya Jegatheesan
Rupalee Patel
Audra Pritt
Dongli Song

JAYPEE

Contents

Section 1: Introduction to Newborn Medicine

1. **Evidence-based Neonatology in the Age of Information and Big Data** 3
   Balaji Govindaswami
2. **Historical Understanding and Current Approaches in Neonatology** 12
   Sharyn Frentner, Robin Daughters Wu, Balaji Govindaswami
3. **Educational Interventions to Improve Newborn Care** 21
   Emily Altick Hartford
4. **Managing Your Patients' Data with an Electronic Database** 31
   Joseph Schulman
5. **Quality and Safety** 46
   Aarti Raghavan

Section 2: Healthcare Services Delivery

6. **Mental Health and Emotional Well-being in the NICU: Addressing the Next Frontier in Family-integrated Care** 59
   Sangeeta Mallik
7. **Utilization of Advanced Practice Providers in the Neonatal Intensive Care Unit** 65
   Valerie D Phebus, Korinne S Van Keuren
8. **Pediatric Hospital Medicine and its Evolution in the United States** 70
   Susan Flesher
9. **Home Follow-up of the High-risk Infant** 74
   Rupalee Patel, Adebola Olarewaju
10. **How Charitable Giving can Enhance Newborn Care** 85
    Elizabeth M Nielsen

Section 3: Perinatal

11. **Obstetric Contributions to Neonatal Morbidity and Mortality—Overview** 95
    Francesca C Ianovich, Matthew J Garabedian
12. **Disorders of the Membranes, Placenta, and Cord** 104
    Shannon Dralla, Matthew J Garabedian
13. **Neonatal Encephalopathy: An Obstetric Perspective** 112
    Stacy Yadava, Matthew J Garabedian

Section 4: Normal Newborn Care

14. **Newborn Physical Examination** 121
    Cherry C Uy
15. **Caring for Families through Pregnancy Loss or Death of an Infant** 129
    Donna F Wallerstein

16. **Circumcision** 136
Stephen Harris

17. **Couplet Care and Common Nursery Issues** 145
Priya Jegatheesan, Sudha Rani Narasimhan, Dongli Song

Section 5: Disorders of Structure and Function in the Fetus and Neonate

18. **Prenatal Diagnosis** 155
Robert L Wallerstein

19. **Birth Defects** 160
Robert L Wallerstein

20. **Inborn Errors of Metabolism and Newborn Screening** 166
Gregory M Enns

21. **Common Surgical Conditions of the Neonate** 177
Stephanie D Chao, Julie R Fuchs

22. **Point-of-Care Testing in Newborn Screening: Hearing Loss and Critical Congenital Heart Disease** 191
Rebecca Barnett

Section 6: Cardiorespiratory Disorders

23. **Extracorporeal Membrane Oxygenation in the Neonate** 205
John Patrick Cleary

24. **Assessment and Treatment Options for Neonates with Congenital Heart Disease** 211
Christina T Sheridan

25. **Pulmonary Hypertension in the Neonatal Intensive Care Unit Setting** 223
Yueh-Tze Lan

26. **Patent Ductus Arteriosus** 235
Priya Jegatheesan, Balaji Govindaswami

27. **Common Cardiorespiratory Disorders at Birth** 243
Kamakshi Devarajan, Balaji Govindaswami

Section 7: Fetal and Neonatal Brain Development, and Neurodevelopmental Follow-up

28. **Feeding and Brain Development in Preterm Infants: Central Pattern Generation and Suck Dynamics** 255
Steven M Barlow, Austin O Rosner, Dongli Song

29. **Feeding and Brain Development in Preterm Infants: Role of Sensory Stimulation** 264
Steven M Barlow, Austin O Rosner, Dongli Song

30. **The Gut-Microbiota-Brain Axis: Implications for Neonatal Neurodevelopment** 275
Dongli Song, Laishuan Wang

31. **Neonatal Seizures** 284
John Sum

32. **Newborn Populations at Risk for Adverse Neurodevelopmental Outcomes** 292
Ira Adams-Chapman

33. **Late Preterm Infants, Cerebral Palsy, and Hypoxic-Ischemic Encephalopathy** 305
Ira Adams-Chapman

Section 8: Growth, Lactation and Nutrition

34. Body Composition and Electrolytes **321**
Sonya Misra

35. Macronutrients and Micronutrients **331**
Sonya Misra

36. Stages of Nutritional Support **345**
Sonya Misra

37. Human Milk Composition and Lactation **355**
Sudha Rani Narasimhan, Alganesh G Kifle

38. Necrotizing Enterocolitis **365**
Arwin Valencia, Antoine Soliman

Section 9: Pain and Addiction

39. Newborn Pain: Recognition and Management **377**
Matthew JR Nudelman

40. Introduction to Addiction Medicine **390**
Balaji Govindaswami

Section 10: Other Common Newborn Conditions

41. Skin Disorders in the Newborn **403**
Noah Craft, Zachary Max Goldstein, Ki-Young Yoo

42. Common Orthopedic Problems in the Newborn **423**
Robert M Bernstein, Ronald A Roiz

43. Oxygenation, Oxygen Saturation, Retinopathy of Prematurity and Other Hyperoxia-related Damage in Newborn Infants: A Return to the Basics **432**
Augusto Sola, Lily C Chen

44. Neonatal Gastrointestinal and Liver Disease **441**
Christopher Fink

45. Nephrology in the Neonate **451**
Dechu P Puliyanda, Rashmi Kirpekar

Index *465*

CHAPTER

4

Managing Your Patients' Data with an Electronic Database

Joseph Schulman

JAYPEE BROTHERS

ABSTRACT

This chapter describes how to think about organizing, storing, and retrieving patient information by means of an electronic database—implemented as an electronic medical record system; and how to evaluate the success of the endeavor. Such NICU systems leverage human cognition, speed workflow, and improve patient safety and clinical outcomes in relation to the degree they succeed in capturing the pertinent knowledge and work processes consistent with optimal information management and human factors. Since both the knowledge of basic science underpinning neonatal care and of how to apply it to patients continually advances, the cognitive tool for managing the information of the NICU is a work always in progress.

INTRODUCTION

The aim of this chapter is to provide an overview of essential concepts for organizing, storing, and retrieving information about patients in a neonatal intensive care unit (NICU), and for reflecting on the success of attempts at implementing those ideas. The following discussion derives, with permission, from previous overviews[1,2] and sources providing greater detail.[3,4]

The daily work of the NICU is too complicated and demanding for a provider to rely on rummaging through a traditional paper-based patient chart to answer a particular question. Such records may not provide much guidance in finding exactly where the required information may lie, nor can they reliably answer a specific question that may arise. A central aim of patient information management is to ensure that one retrieves the required information quickly; ideally, in the first place one looks for it. But that is far from all that a NICU patient information system should aim to achieve.

In conducting the daily work of the NICU, we often deal with overwhelming amounts of information. To some extent, we recognized this information overload long ago, when we set up paper-based information systems—the traditional medical chart. But, a paper-based system is inadequate.[5,6] In order to recall a laboratory value or remember to check on an X-ray, we may jot the information down on an index card or the back of an envelope. Even better, so we do not lose it, we might write it on the leg of our scrub pants. The problem becomes much larger when we try to learn from past experience. Systematic review of previous patients is labor intensive. Provider notes in paper-based charts commonly are free-form and free-text. That is, there are no strict rules about precisely what must appear in each note; the user can write anything at all (within a size limit). People newly introduced to computers tend to like free-text entry because they believe this "gets all the data into the computer". Unfortunately, typical database query programs cannot reliably search this kind of data. Nor can one be certain that what was entered in free text amounts to an accurate, apropos, clear, coherent, and complete description of the notion to be captured. This potential variability and inconsistency can crucially limit accurate and robust inference and new learning. As a result, chart reviews cannot be counted on to inform unbiased and unconfounded learning from experience; relevant factors may be omitted.

Then, there is the issue of workflow. It takes much time to meticulously work through large amounts of information, filter what is pertinent, reflect on it, draw inferences, devise a plan, implement it, and document everything. Even today, many medical providers need a tool that can speed up their workflow and give them greater mastery over the data with which they work.

A computer-based information system is potentially such a tool. Several vendors of electronic medical records now offer a tool for the NICU. For readers who work in institutions that provide such products, this chapter aims to provide a framework for understanding and evaluating such a tool's performance. However, only a fraction of neonatologists currently can take advantage of such tools, so this chapter's aims are broader still—to provide an overview of electronic database design and implementation sufficient to guide readers who wish to create their own NICU tool.

HOW WE KNOW: TACIT, IMPLICIT, AND EXPLICIT

A tool that leverages a NICU provider's mastery of patient information may require profound advances in several key areas of work: how an organization's members collect, manage, and interpret data; how they learn from their experience; and how they share the learning. These advances may require fundamental changes in how providers think about their work and their patients, namely, changing from tacit or implicit thinking to explicit ways of thinking.

Tacit knowledge tends to be hard to communicate to another person, either through speech or writing it down. Examples of tacit knowledge include how to ride a bicycle, how to play the piano, and how to be a neonatologist. Although one can read a textbook on any of these skills, the explicit knowledge communicated will fall short of all that must be known to achieve authentic competence. Explicit knowledge can be written down, communicated, and understood by another person. For example, identifying that a neonate has a serum sodium concentration of 123 mEq/L and is therefore hyponatremic, is explicit knowledge. Implicit thinking and knowledge entail unspecified elements and algorithms; in certain settings, some may consider this to be intuitive thinking. For example, an NICU may have an implicit policy for screening visitors who may have a communicable illness. That is, staff may know that they need to ask visitors about this concern, but no written document exists that explicitly articulates exactly who asks a visitor exactly what questions, or exactly when.

Unless a NICU process is written down—explicitly mapped or meticulously described in words, it may well be implicit. Consider, for example, the process of intubating a neonate. There are many details that might be specified, including the particular staff involved, exactly what each one does (consider precisely in how much detail this should be done), the supplies and medications to be made available, and exactly how the endotracheal (ET) tube is placed. Again, consider precisely how much detail could be specified, from opening the neonate's mouth, to positioning the tongue, to how the laryngoscope is held and moved in response to what is visualized, to exactly how the tube is advanced in response to what is visualized, to how the laryngoscope is withdrawn from the oropharynx, to exactly how the ET tube is secured, and exactly how it is determined that all went as planned. The point is that dozens and dozens of component steps may rush by us in an instant of time. Humans cannot attend to all this detail. If questioned about it after the fact, how can we formulate answers from a blur?

Paper-based patient information systems allow providers to dodge this troubling reality of NICU care, because such systems generally do not force us to engage with the gap between what we attend to and what must be entered into the data storage system, if it is to genuinely serve our needs. Paper-based patient information systems typically allow providers to decide what to include in a progress note. As a result, such systems allow workers to avoid facing the challenge of transforming the imprecise, tacit, and even ad hoc ways of thinking that have underpinned so much of healthcare, into more useful collections of data and information about our work. This is also true of computerized systems that do not facilitate explicitly identifying and mapping all important aspects of daily work. With paper-based or underconceptualized computerized systems, there is no system other than retrospective review to detect omitted pertinent information.

The important point is, it is only when we carefully reflect on our work and the limitations of paper-based and other underconceptualized patient information systems that we may appreciate that we do not have total mastery over what unfolds in the NICU; and that we could do much better than we currently do.

REPRESENTING THE WORK OF THE NEONATAL INTENSIVE CARE UNIT IN A COMPUTER

A computer is a wonderful tool to keep track of more things than we can in our heads or on paper. But, most computers and the programs they run are not designed to handle tacit knowledge and implicit processes in a way that leverages human competence. If we are to benefit from the things computers can do enormously better than humans can, we must think explicitly about clinical data elements. This entails articulating precise methods for collecting,

JAYPEE BROTHERS

managing, analyzing, interpreting, and acting on this data in excruciatingly fine-grained detail. Unfortunately, in their desire to gain end-user acceptance, institutional decision makers may replicate in the new computer technology the long-standing tacit and implicit methods, thereby limiting and obscuring the potential otherwise available. This may well represent an innocent misjudgment. Few clinicians are solidly trained in data modeling, database design, and implementation.[7] Optimized information management tools require designers who can straddle the domains of clinical care, information technology, and data analysis.

Often when clinicians imagine a computer managing their patient information, they imagine the computer is doing it the same way the clinicians have always done it; just faster, and with less direct clinician involvement. But, we cannot consider the computer a "black box" that will do our work if only we enter the kind of unspecified free-text as we did in the days of paper-based charting. A computer is unintelligent; it only does what it is programed to do. If you had to tell a layperson exactly what to do with some data so that it was properly processed, could you? Well, that is all we are doing when we use a computerized patient information system—we tell a software application exactly what data/information are important, where to store it, how to manipulate it, how to display it, so that we can tell an accurate, apropos, and reasonably complete story about our patient; one that will also inform learning from experience. That sounds straightforward enough, until one reflects on the precise operational details to carry it off.

To conduct our work with a computer, we must precisely specify as many essential details about our work as possible. We must meaningfully represent our work on a computer and capture the essentials, carefully map conceptual elements to precise software elements, and specifically structure it. Often, more than one representation option is possible, yielding varying degrees of success for the performance of the resulting tool. That tool, in one form or another, is a database software application.

Microsoft® Access can serve very well for readers who wish to create such a tool for their own NICU. This software application is quite adjustable, especially because it allows one to write programing code to customize features and even to create new features. However, it takes a vast amount of cataloging and narrative to communicate all the necessary information. So, readers should temper their expectations about becoming a powerfully competent Access user (power user) simply by reading the present chapter or even a book devoted to the subject.[3] What this chapter can do is describe the components of the task and provide assurance that success can indeed be achieved over time.

At its simplest, a database is a collection of data. The term does not necessarily imply computers or software program applications. The card catalog (increasingly replaced by a computer) in many public libraries is an example of a useful, low-tech database. Importantly, computerized, high-tech databases are not just faster versions of low-tech databases. Computerized databases usually reflect more sophisticated design principles than traditional, low-tech databases.

If the Database is the Solution, What is the Problem(s)?

One fundamental design principle for developing a useful database is to articulate clear ideas about the problems it is intended to solve and the goals to be achieved by the product of the software development.

The Institute of Medicine proposed explicit aims for patient records:[6]

- Support patient care and improve its quality
- Enhance the productivity of healthcare professionals and reduce the administrative costs associated with healthcare delivery and financing
- Support clinical and health services research
- Be able to accommodate future developments in healthcare technology, policy, management, and finance
- Keep mechanisms in place to ensure patient data confidentiality at all times.

These are explicit, but broadly articulated. To achieve them, we require aims for the individual components of patient records, described at a fine-grained level of detail. Many clinicians have not thought pointedly about exactly what an admission note, or a progress note, or a discharge summary, should achieve. Further, how many readers can operationally define what they mean by quality of care?

Ontology

Even if we share a common idea of information and how to explicitly represent it, we may still have problems communicating. The term ontology describes a special kind of framework to represent our shared knowledge. Ontology in our context is how we specify our idea of our work. It includes precisely defining the vocabulary we use in our discourse, and in our databases. On their own, workers in different settings come up with different terms and structures for representing essentially similar information. For example, consider the variety of definitions of chronic lung disease, an entity often represented in a database as a simple "yes" or "no".

The lack of a common ontology has been called the "Tower of Babel" problem.[8] Within different databases, data with the same label may have different meaning. Other

JAYPEE BROTHERS

times, we may mean the same thing but use different labels. The potential for learning from our aggregate experience is awesome, but it requires that we resolve the inconsistencies among our data tools. Uniform terminology, representation, and data structure are central to achieving the potential of the new information technology.[9] Calibrating all our data models and database applications to one common ontology could produce unimaginable opportunity to do our work better. To learn more about uniform terminology for healthcare records, visit the websites of the standards-development organizations, called Health Level 7[10] and Systematized Nomenclature of Medicine—Clinical Terms.[11]

Data Modeling

Explicitly structuring the data that constitute our work entails a data model. A data model is an abstraction aimed at broadly representing the ideas and things that constitute an organization's work. It is the framework that specifies what kind of data to keep and how to store them. A data modeler works with information system users much as an architect works with a building's future dwellers.[12] Both architects and data modelers are designers, people who work with problems that have more than one correct solution.[12] One models—maps—the important objects and events of the reality, so they may be "saved" and subsequently "manipulated".

Why provide technical detail on this subject? Well, if you retain an architect to design and oversee construction of your home, would you not want to understand and approve the blueprints? Would you just move in without repeatedly confirming that the design and the product represent your residential needs?

We shall concentrate on a particular type of data model called the relational data model, so named because it is based on the notion of mathematical relations—for our purpose, tables containing *x* rows and *y* columns of data. Though tables often are related to each other, that is not why the model is called relational. The relational model specifies a variety of table features. Each of the relations or tables must have a unique name, as must each of the columns or attributes. The values each attribute may have are specified by a domain. This notion of domain introduces meaning to the data contained in an attribute and helps to avoid incorrect relational operations. For example, a domain might dictate that telephone numbers contain only digits and may not be subjected to arithmetical operations. Each row (observation and record) in a table must be unique. Each cell, that is, each intersection of a row and column should contain only one value (jargon: field values should be atomic). In other words, a single cell holds a single answer. Trying to query cells containing multiple answers or free-text may be problematic.

What is a Query, a Table?

A database query is a question you formally ask of a database. You ask the question operationally, providing a set of instructions for finding a subset of the data in the database. To produce a report, for example, a daily progress note from the data that were entered into a variety of fields, a number of queries must be run. Query results and answers are assembled according to precise instructions specifying exactly where and how they are to appear in the report. Queries yield answers by means of relational operations stemming from mathematical set operations. Queries to a relational database commonly run via a special programing language called structured query language.

The notion of a table is a key concept. A table is a container for storing data that share common attributes. Tables have rows (horizontal divisions) and columns (vertical divisions). Readers who are familiar with Microsoft® Excel's rows and columns are thus familiar with this notion of a table. Each column describes one attribute of the more overarching idea the table is intended to describe. Each row contains one instance of the table's attribute set, one observation of the thing the table describes. Each row is also called a record. Each column is also called a field.

If you had a table for storing several attributes of your patients, i.e. a "patient's table," each row would contain the information for one patient (one record), with each column recording the information for each attribute (field: for example, birth date, birth weight, gestational age, and so on). For reasons beyond the present scope, it is generally not feasible to store all the pertinent information for a patient in the NICU in a single table, so a core challenge is to identify the pertinent tables needed, and the pertinent attributes for each table.

Relationship Diagram

The logical connection between information in one table (relation) with the information in another table (relation) is called a relationship. When one record in table A can relate to only one record in table B, a one-to-one relationship exists. For example, each patient can have only one set of admission vital signs because a second set would no longer describe the condition at admission. When one record in table A can relate to many records in table B, a one-to-many relationship exists. One mother, for example, can have more than one infant.

Some appreciation of what data modeling is about may be gained by looking at a diagram of a relatively simple NICU data model. **Figure 1** shows the data model for electronic neonatal intensive care unit (eNICU), a software tool for managing patient information in the NICU.[3,13] The figure is very complicated (yes, I also said that this

JAYPEE BROTHERS

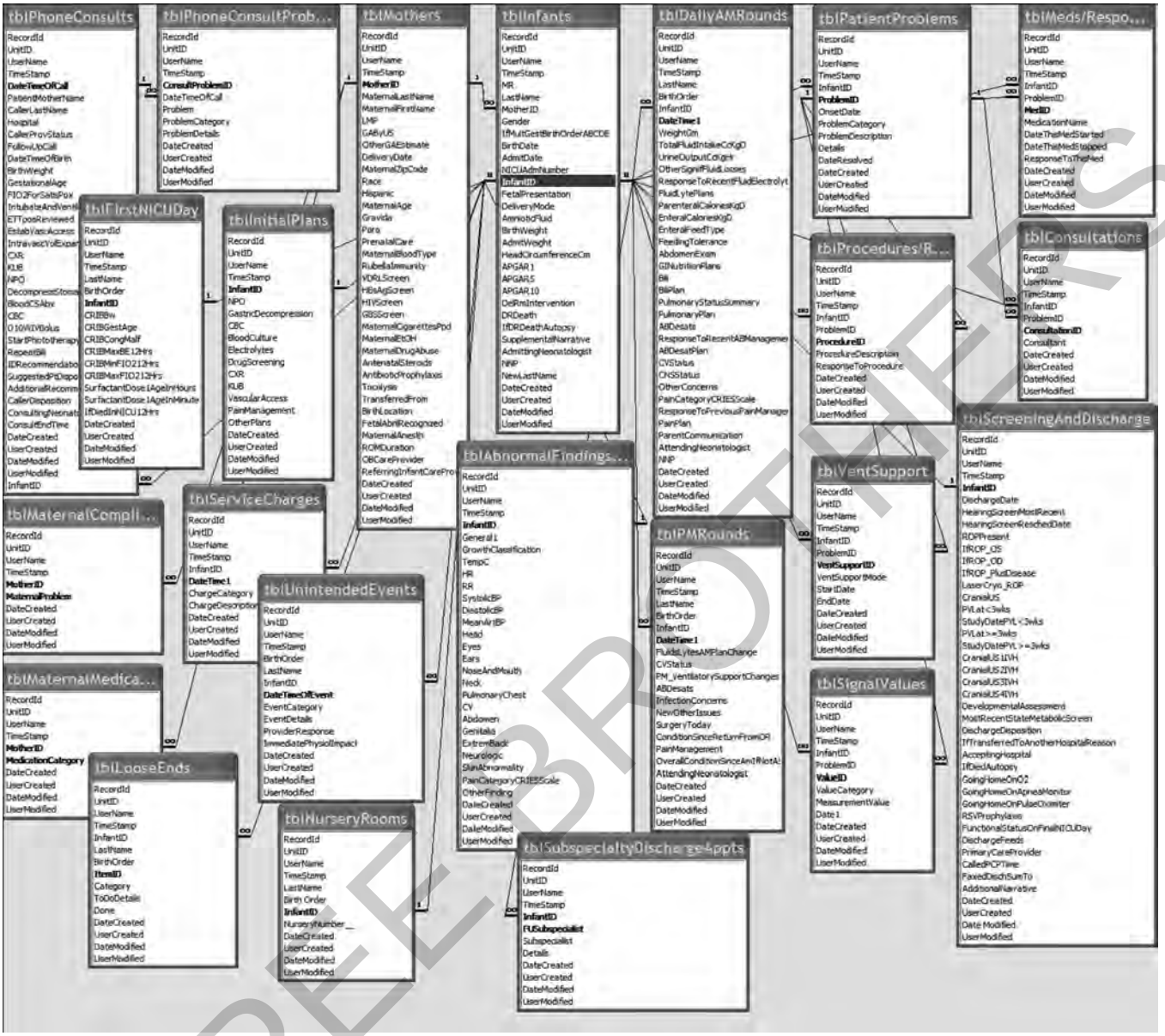


Fig. 1: Data model of electronic neonatal intensive care unit. From Schulman[3] with permission.

is a relatively simple NICU data model), but it makes the point that someone ultimately must decide which data elements to include and how to connect them—how they relate. Accepting a data modeler's design without careful reflection on how well the design has mapped the daily work, i.e. the users' (not the modeler's) reality, can result in a disappointing product. Moreover, it is important to note that the disappointment may not be apparent for some time. It may require the accumulation of many records and repetitive querying to discover suboptimal or problematic function.

Figure 1 uses the convention of naming a table with the prefix "tbl" and "camel caps" instead of a space between words that constitute a name. Relationships between tables are indicated by lines drawn from the primary key field, in bold font, of one table to the foreign key field of the other table. When one or more fields uniquely identify each record in a table, such a field is called a primary key. That is, for each record, the value entered in the primary key field(s) is unique among all records in the table. Without a primary key, records in a database may become confused and database content degraded. A good way to link a record in one table with a record in another table is for each record to share some common attribute value. Thus, if we wish to connect a particular record in a table of infant data with a particular record in another table of maternal data, we would ensure the infant data table includes a field containing the mother's unique identifier—primary key

JAYPEE BROTHERS

value. Such a linking field is called a foreign key. Together, the linked records describe one instance of a higher entity, the mother/infant dyad in our example, constituted by the various tables in aggregate.

When the relationship is one-to-one, the number 1 appears at each end of the connecting line. When the relationship is one-to-many, the many side of the relationship is represented by the symbol ∞. Occasionally, a relationship may be many-to-many. In **Figure 1**, each patient may have many problems, each of which may be treated with many medications. Similarly, many medications may be used to treat many different problems.

More concretely, consider an infant who experiences multiple episodes of nosocomial infection, each instance treated with the same combination of antibiotics. In a relational database, we cannot directly model the two tables in a many-to-many relationship. Each of the two tables must exist in a many-to-one relationship with a common linking table. In **Figure 1**, tblInfants, tblMeds/Responses, and tblPatientProblems conform to this idea, but this is not readily apparent. **Figure 2** depicts the relationship without the distractions. A many-to-many relationship exists between tblMeds/Responses and tblPatientProblems. That relationship is implemented by using tblInfants as a linking table, in a one-to-many relationship with each of the other two via the InfantID field. At another conceptual level, tblPatientProblems and tblMeds/Responses have a separate one-to-many relationship via the ProblemID field. This relationship models the idea that for each problem, many medications may be used.

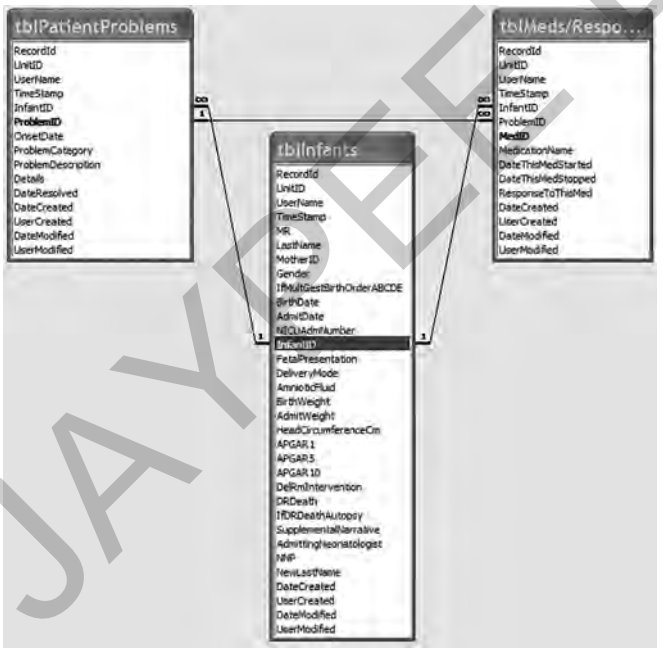


Fig. 2: Relationship diagram: tblInfants is the common linking table for the many-to-many relationship between tblPatientProblems and tblMeds/Responses. Each listed attribute represents a table column. The one side of a relationship is denoted by the number 1 at the end of the connecting line. The many side of the relationship is denoted by the symbol ∞. From Schulman[3] with permission.

One of the tables shown in **Figure 1** is labeled tblSignalValues. Via the ProblemID field, this table is in a many-to-one relationship with the ProblemID field in tblPatientProblems. As just discussed, the latter table is in a many-to-one relationship with tblInfants. The tblSignalValues was designed to store information about particular laboratory studies, those results that signaled meaningful discrimination contributing to establishing a specific patient problem. As such, the model deals with the challenge of ensuring that laboratory information is not merely automatically placed in the patient record without assurance that the information has been processed by the appropriate professional, filtered, and incorporated in decision-making. Thus, the model connects particular laboratory results and specific medications with a particular problem.

The eNICU data model describes, but does not explain. Exactly how did I decide I needed a separate table for infants, patient problems, loose ends, etc.? Truth be told, the scheme did not emerge from one exhilarating epiphany. It is the result of a reiterative process. That process reflected guidelines ranging from simple rules, such as provide a separate table for each class of "real world" objects about which you are trying to store information in the database, to complex ones, such as those that concern normalization. Normalization amounts to a set of design rules specifying what each of the multiple tables in a database is about, and the attributes that belong with each table. These rules generally optimize data storage and retrieval by anticipating the things you will want to do with the data, and ensuring that you will be able to carry them out. By this means, normalization enables reliable queries, i.e. queries that reliably provide the correct answer to a question every time you ask the question. This is a crucial point—if one runs the same query 100 times in an improperly conceived database application, the same result may not appear all 100 times.

A Cognitive Tool for Clarity and Focus

One aspect of the data model in **Figure 1** merits special consideration: the many fields and associated categories that may pertain to an infant. Readers may well ask, "Does the fact that a computer does a better processing job using explicit methods necessarily make it the preferred cognitive tool for a human being in the NICU who is accustomed to tacit thinking?" In 1964, Marshall McLuhan noted that "when International Business Machines Corporation

JAYPEE BROTHERS

(IBM) discovered that it was not in the business of making office equipment or business machines, but that it was in the business of processing information, then it began to navigate with clear vision".[14]

Neonatal care providers perform many procedures, but at the core, they must usefully and accurately categorize a patient. The medications provided, the fluid intake, the nutritional support, the respiratory support, the placement of a vascular catheter or a chest tube, and even what we say to the parents, all derive from recognizing that a neonate's profile places that infant in particular categories, and categorical particulars inform what is done for that neonate. Examples of such particulars include elevated pCO_2, free water excess, hyponatremia, and growth restriction. Database fields that are not concerned with such categorization still generally support the process. For example, such fields describe what is to be, or was, done for a patient as a result of being in a particular category, or identify the patient; or describe when something happened, or describe a patient exposure or characteristic relevant to establishing a category, or an outcome. Thus, free-text fields provide no advantage in conceptualizing the work of the NICU unless they serve to describe rare situations or other dimensions of neonatal care that have not yet submitted to explicit representation by data modeling.

Steps in Designing a Patient Database[3]

Step 1

State what the database is to achieve. For example, "This NICU database will maintain a core data set that attending neonatologists, neonatal nurse practitioners, and residents use in day-to-day patient record documentation and for NICU evaluation".

Step 2

Specify exactly what you want to accomplish with the data.

a. List specific tasks the database will support, for example:
   i. Produce admission notes
   ii. Produce daily progress notes
   iii. Automatic attending sign-out summary
   iv. Report service charges
   v. Produce discharge summaries
   vi. Satisfy organizational chart audit requirements
   vii. Populate the fields required for other database projects
   viii. Support NICU-level evaluation.
b. It should be appreciated that all above-mentioned points (i-viii), each require specification of fine-grained detail.

Step 3

Describe the current reality.

a. At present, how do we collect data (forms, index cards, software application interfaces, etc.)?
   i. Collect samples of each way you currently record data:
      - For each, describe how it is used and for what purpose.
b. How do we currently present the information?
   i. Collect sample chart notes, etc.
c. What information do users seem to need that they do not currently have?
   i. Why do we need it?
   ii. How do we know we need it?
   iii. What will be different if we had it?
   iv. What activities or documents rely on it?

Step 4

Make a list of all fields and calculations gathered from Step 3 (a and b):

a. This is a preliminary field list; it contains current fundamental data requirements of the NICU and is the starting point for the design of the new database.
b. Identify apparent duplicate fields:
   i. Do they represent the same attribute of the same entity?
      - If so, drop all but one
      - If not, rename all but one so each uniquely describes a particular dimension of the entity.
c. Also look for fields with different names that actually describe the same attribute. Drop all but one.
d. Place every calculated field (a field derived from other fields) on a separate calculated field list.

Step 5

Ask for feedback on the list by all users. To encourage participation, schedule meetings by mutual assent and provide an appealing environment, e.g. offering refreshments.

Step 6

Create the structures to contain the data.

a. Tables:
   i. Consider which fields appear to belong together:
      - Associate each field from the field list in step 4 with an appropriate table.
   ii. Review each table to ensure that it:
      - Represents only one thing, or entity (object, event, or classification)
      - Contains no duplicate fields.

JAYPEE BROTHERS

iii. Describe in writing what each table represents and exactly how it contributes to what the database is to achieve. Return to this description days or weeks later and assure yourself what you wrote is clear and coherent. Next, ask for a coworker's opinion of it.
iv. Edit the table names:
- Use unique, descriptive, and plural names that make sense to all users.

b. *Fields (attributes):* Assure yourself that each field is indeed an attribute of the object, event, or classification the table represents; relocate or delete fields as you think appropriate. Edit the fields. Use unique, descriptive, and singular names that make sense to all users. Check that each field is designed to contain only a single (atomic) value. Remodel a multivalued field as a discrete table. Designate the primary and foreign keys to link the original table to the new table. Identify each field that serves to link two tables; check that it appears in both tables, albeit under different respective field names.

Step 7

Create a table on paper using "dummy" (fabricated) data and look for anomalies.

Step 8

Continue to refine the table structures:
a. Aim for redundant data only in linked fields.
b. Aim for duplicate fields only in linked fields.

Step 9

Designate keys for each table:
a. Start by identifying all candidate keys.
b. Next, select one primary key per table:
i. Designate the remaining candidate keys as alternate keys.

Normalization: Objectives and Strategy

Several of the above database design guidelines endeavor to structure the data to conform to a branch of mathematical set theory called normalization theory. Normalization is a process applied to a set of relations so that:

- Queries that logically may be asked of a set of relations indeed can be asked and will be answered correctly
- Relations store a minimum of redundant data.

An informally determined set of relations (tables) that is not normalized may be incapable of handling all possible queries. Further, such a set of relations may take up more storage space than necessary and therefore, the application will perform slower, increasingly so as the number of patient records grows. As previously mentioned, it may even provide inaccurate query results! Thus, arbitrarily constructed databases, though they may contain the data elements individually deemed appropriate for representing the daily work, when interrogated may function ever more slowly, yield misleading or wrong answers, or at times fail.

Normalization assures the functionality of a database design and provides a nonarbitrary method for determining the appropriate tables. Absent normalization, the patient information management system sooner or later will fail as a tool to facilitate the work of the NICU. Normalization entails applying a series of tests to a group of tables. You apply the tests at successively more detailed levels of scrutiny. Each level imposes greater restrictions on the tables, ensuring greater resistance to problems with data management. Normalizing to three levels usually provides satisfactory performance results. Four more levels may be applied, but these deal with situations that most people are exceedingly unlikely to create, even accidentally.[15,16]

Implementation of a Database Design

Arcane and intimidating as the preceding overview of database design may be to an individual new to information technology, it covers only foundational aspects of a computerized patient information management system. A selected database design must then be implemented as a software application by which workers can achieve their information management goals. Implementation of a database design is achieved within a general database software platform such as Microsoft® Access, or within a more specific electronic medical record software platform developed by commercial vendors such as Epic, Cerner, and GE, to mention a few.

Implementation entails actually programing into existence the components of the database software application, beginning with the designed relations, the tables. So-called forms are a means for entering data into tables, and also for viewing data already in tables. Forms enable users to add to, or examine what is already in one or more tables without putting at direct risk what already resides in the base tables. Queries pull the required data from the fields in tables. Queries amount to asking questions operationally, by providing a set of instructions for finding a subset of the data in a database. Reports configure and display the results of the various queries to produce products such as admission notes, progress notes, hand-off updates, and discharge summaries.

Figure 3 illustrates a discharge note constructed by the eNICU software for a fictional patient. It reflects data stored in the model described in **Figure 1**, and is an example of but one of several possible reports that are constituted by results from particular queries that run automatically when requested for a particular patient.

JAYPEE BROTHERS

eNICU: Discharge Note

Last Name: New Last Name Female

MR 123123
Infant ID: 1231231
AMC Adm No 1

GA *LMP* 35 *U/S* *Other est.*

Birth Date 9/25/2002
Admit Date 9/25/2002
Discharge Date 9/30/2002

Amniotic Fluid clear
Fetal Presentation transverse
Delivery Mode Caesarian
DelRm Interventio BMV

APGAR 1, 5, 10 6 7 8
Birth Weight 1475
Admit Weight 1410
HC (cm) 30

Supplemental narrative at admission:
Concern for GI perforation.

Maternal Information

OB Care Provider Dexter, S.
Referring Infant Care Provider

Delivery Date 9/25/2002 *Transferred From* Bassett (Cooperstо *Birth Location* AMCH

Last Name Test3 *Age* 19 *Gravida* 2 *Para* 0010 *Prenatal Care* Y

First Name Jane *Race* White *Hispanic* N

LMP 1/23/2002 *Smokes? (ppd)* 2

GA by U/S *Alcohol use?* N

Other GA test. *Illicit drug use?* N

Fetal Abnl Recognized N *Anesthesia* spinal/epidural

Prenatal Screening

Blood type	*Rubella*	*VDRL*
AB+	immune	negative
HBsAg	*HIV*	*GBS*
negative	negative	unknown

Maternal Problem

Asthma
Depression (meds)
Poor maternal weight gain

Medication Category

antidepressant
bronchodilator

Tocolysis N
Antenatal steroids? Incomplete (<1 or >7 days PTD
ROM Duration 1-12 hours
Abx prophylaxis? none

Problems/Interventions

1. *Category*	*Description*	*Details*	*Onset*	*Resolved*
CNS	Neuromuscular tone: low		9/25/2002	9/27/2002

Ventilatory support associated with this problem

	Start Date	*End Date*
conventional mech ventilation	9/25/2002	9/26/2002
HFOV	9/26/2002	9/26/2002
NCPAP	9/26/2002	9/27/2002
oxyhood	9/27/2002	9/28/2002

JAYPEE BROTHERS

Procedures associated with this problem
9/25/2002 10:28:21 AM LP — Subseq status unrelated

Consultants for this problem
Neurology

2.	*Category*	*Description*	*Details*	*Onset*	*Resolved*
	INFECTION	sepsis < 72 HRS: suspected		9/25/2002	9/28/2002

Medications associated with this problem

	Started	*Response*	*Stopped*
Ampicillin	9/25/2002	improved	9/27/2002
Gentamicin	9/25/2002	improved	

Signal values related to this problem
WBC 37 9/25/2002

Procedures associated with this problem
9/25/2002 10:40:38 AM LP — No detectacle effect

Consultants for this problem
ID

3.	*Category*	*Description*	*Details*	*Onset*	*Resolved*
	PULMONARY	RDS		9/25/2002	9/30/2002

Medications associated with this problem

	Started	*Response*	*Stopped*
O2	9/25/2002	improved	9/28/2002

Signal values related to this problem
pCO2 96 9/25/2002

Procedures associated with this problem
9/25/2002 10:37:22 AM Endotracheal intubation — improved

4.	*Category*	*Description*	*Details*	*Onset*	*Resolved*
	PULMONARY	O2 requirement		9/25/2002	9/30/2002

Signal values related to this problem
pO2 32 9/25/2002

JAYPEE BROTHERS

5.	Category	Description	Details	Onset	Resolved
	SURGICAL	Jejunal atresia		9/25/2002	9/30/2002

Procedures associated with this problem

9/25/2002 11:03:11 AM jejunostomy Improved

Screening and Discharge Information

Discharge Date 9/30/2002

HearingScreen (most recent) Passed ***If Screen Re-scheduled, Date:***

ROP present? Yes ***If ROP_OS:*** Stg 2, Zn 1 ***If ROP_OD:*** Stg 2, Zn 2 ***Plus Disease?*** O.S. and O.D

Laser Cryo_ROP: O.S. and O.D.

Cranial US: Y

IVH stage on cranial U/S
#1 0
#2 1
#3 1
#4 1

PVLat<3wks: N ***Study Date PVL<3wks:***

PVLat>= 3wks: Y ***Study Date PVL>= 3wks:*** 10/30/2002

Developmental Assessment Normal

Functional status Normal newborn

Most recent metabolic screen Normal ***If died, autopsy***

Discharge Disposition Home ***If transferred, reason*** ***AcceptingHospital:***

Discharge equip/supplies: ***O2*** Y ***Pulse Oximiter:*** Y ***Apnea monitor*** Y

DischargeFeeds: Breast milk + HMF ***RSV Prophylaxis:*** Indicated:<32 wks

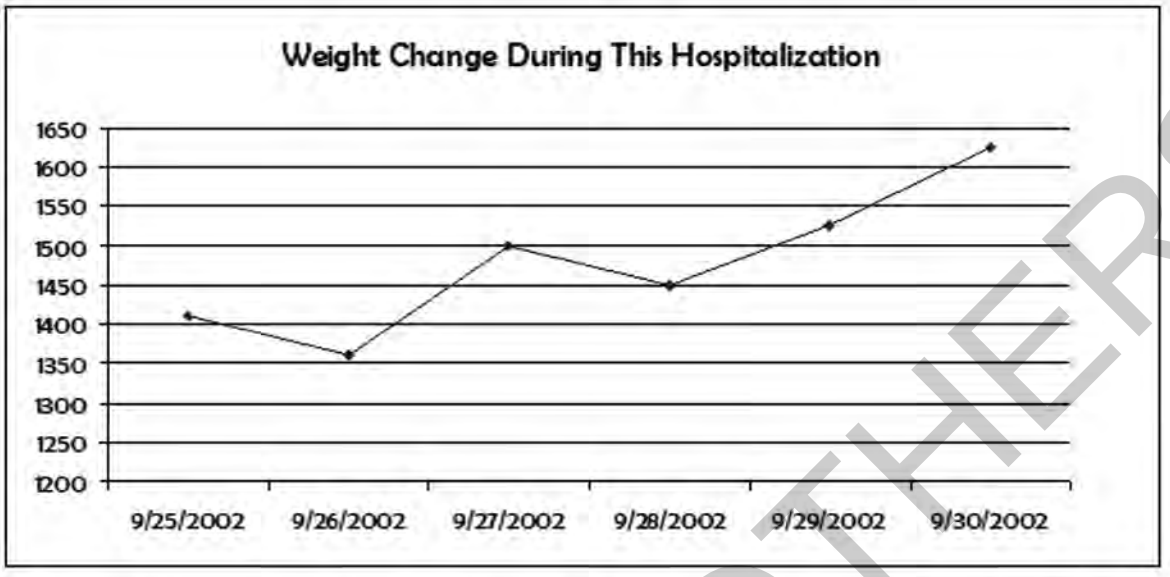
Primary care provider Tomiak ***Discussed:*** 9/30/2002 10:00:00 AM ***Faxed copy to:*** (518) 262-5421

Pre-discharge exam unremarkable except as noted below:

Pertinent Findings on Discharge PE and Supplemental Narrative

Normal exam for age

JAYPEE BROTHERS



Care Process Totals

Oxyhood
Start Date: *End Date:*
9/27/2002 9/28/2002

NCPAP
Start Date: *End Date:*
9/26/2002 9/27/2002

Conv Mech Vent
Start Date: *End Date:*
9/25/2002 9/26/2002

HFOV
Start Date: *End Date:*
9/26/2002 9/26/2002

Cum Vent Days: 2
Cum O2 Days: 3
First Surfactant Dose *hrs. min.*: 0 12

Cum Parenteral Nutrition Days: 6
Days to 80 cal/kg/d: 3
First Enteral Feed: 28-Sep-02 breast milk
First Day Off IV: 30-Sep-02 breast milk

On Discharge Day *Weight Gm*: 1625 *Calories/Kg/D*: 86

Subspecialty Follow-up

surgery 2 weeks

Meds Continuing After Discharge

Medication Name	*Date Started*
gentamicin	9/25/2002

New Last Name 123123 *Discharge Summary* *Page 4 of 5*

JAYPEE BROTHERS

Fig. 3: Database software application report: a discharge summary for a fictional patient.
From Schulman[3] with permission.

for detailed information on pediatric books, visit our website www.jaypeebrothers.com, for detailed information on ped

Preventive
Newborn Health:
A global approach in the age of information

Key Features

- A preventive and public health approach to common neonatal problems presented in an innovative and deliberately disruptive style. We review optimal perinatal and cardiorespiratory management and explore surgical, dermatological, and orthopedic disorders
- A primer on neonatal neuroscience and mental health, from understanding addiction and its impact on fetal and infant brain structure and function, to studying long-term neurodevelopment in other high-risk infants
- An introduction to emerging knowledge in the biome and neuroscience, with a focus on nutrition. We also share current healthcare service delivery models in newborn medicine, with pediatric hospitalists and advanced practice providers becoming increasingly important to patient care and research
- A guide to the essential contribution of charitable foundations and like-minded agencies, including patient and family advocates, and their critical importance to augmenting resources for underserved women, children, and families.

Balaji Govindaswami MBBS MPH FAAP is a neonatologist and public health champion who trained at the University of Lagos College of Medicine, serving his British Commonwealth rotating internship at the San Fernando General Hospital in Trinidad. He then studied Maternal and Child Health at the University of Alabama in Birmingham, doing his Pediatrics internship and residency at the University of Kentucky in Lexington, followed by Neonatal-Perinatal and ECMO fellowships at the University of California, San Diego, USA.

He began his teaching career as a graduate assistant at the University of Alabama in Birmingham, and was an instructor in Pediatrics at the University of California, San Diego. He has held faculty appointments at the University of California, Irvine, and Los Angeles, and Stanford University School of Medicine. Board certified in Pediatrics/Neonatal-Perinatal Medicine, he has served Santa Clara County Health and Human Services as chief of neonatology and director of neonatal intensive care for 14 years. He has returned to central Appalachia to serve as Professor and Chair in Pediatric Preventative and Addiction Medicine at Marshall University Joan C Edwards School of Medicine.

A tireless advocate for global health, he has taught in numerous cities in several countries and presently volunteers in Asia and Africa. He has been a Castle Connolly "Top Doc" for 20 consecutive years and has won numerous university teaching and public service awards for Quality, Mentorship, Advocacy and Excellence in Patient Safety and Outcomes. He has published over 150 abstracts, peer-reviewed papers and chapters.

Printed in India

Available at all medical bookstores
or buy online at www.jaypeebrothers.com

JAYPEE BROTHERS
Medical Publishers (P) Ltd.
EMCA House, 23/23-B, Ansari Road,
Daryaganj, New Delhi - 110 002, INDIA
www.jaypeebrothers.com

Join us on facebook.com/JaypeeMedicalPublishers

Shelving Recommendation
PEDIATRICS

ISBN 978-93-5270-473-6